```
File 347:JAPIO Nov 1976-2004/Jun(Updated 041004)
         (c) 2004 JPO & JAPIO
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200463
         (c) 2004  Thomson Derwent

Set     Items   Description
S1    1273593   OS OR OPERATING()(SYSTEM? ? OR ENVIRONMENT? ?) OR APPLICAT-
             ION? ? OR PROGRAM? ? OR SOFTWARE? ?
S2       2273   S1(5N)(REDUC???? OR SHRINK??? OR LOWER??? OR DECREAS??? OR
             TRIM???? OR CONTRACT? OR CONDENS?)(5N)(SIZE OR SCALE OR FEATU-
             RE? ? OR FUNCTION? OR MAGNITUDE)
S3      15271   S1(7N)(SUBSET? ? OR SIMPLIF? OR SMALL??? OR COMPACT? OR MO-
             DULAR? OR FOOTPRINT? ?)
S4       4325   S1(5N)(EXTRACT??? OR PARS??? OR SELECT??? OR PICK??? OR ID-
             ENTIF???? OR IDENTIFICATION)(5N)(COMPONENT? ? OR MODULE? ? OR
             FEATURE? ? OR FUNCTION?)
S5     109053   (APPLICATION? ? OR PROGRAM? ? OR CODE OR SOFTWARE OR FILE?
             ? OR OS OR OPERATING()SYSTEM)(3N)(DEVELOP? OR GENERAT? OR CRE-
             AT? OR DESIGN?? OR DESIGNING OR BUILD? OR CONSTRUCT? OR PRODU-
             CE? ? OR PRODUCTION OR AUTHOR??? OR WRIT???)
S6      17642   S5(3N)(KIT? ? OR TOOLKIT? ? OR PLATFORM? ? OR ENVIRONMENT?
             ? OR TOOL? ? OR PACKAGE? ? OR SYSTEM? ? OR ARCHITECTURE? ? OR
             SOLUTION? ? OR METHOD? ? OR METHODOLOG??? OR MODULE? ? OR FRA-
             MEWORK? ?)
S7         74   S2 AND S6
S8        416   S3 AND S6
S9        421   S4 AND S6
S10       880   S7:S9
S11       703   S10 AND IC=G06F
S12     57992   S1(5N)(COMPONENT? ? OR MODULE? ? OR FEATURE? ? OR FUNCTION?
              ? OR FILE? ? OR LIBRARIES OR LIBRARY OR DLL OR DLLS OR OBJEC-
             T? ?)
S13       479   S11 AND S12
S14        47   S13 AND (OS OR OPERATING()(SYSTEM? ? OR ENVIRONMENT? ?))
S15        29   S14 AND AC=US/PR
S16        25   S15 AND AY=(1970:2001)/PR
S17        27   S14 AND PY=1970:2001
S18        37   S16:S17
S19     42390   (TRACE? ? OR TRACING OR RESOLV? OR RESOLUTION? ? OR CONNEC-
             T? OR LINK???)(5N)(REFER??? OR REFERENC? OR DEPENDEN? OR ASSO-
             CIAT? OR CORRELAT? OR RELATED)
S20         9   S11 AND S19
S21       115   S2:S4 AND S19
S22        12   S21 AND (OS OR OPERATING()(SYSTEM? ? OR ENVIRONMENT? ?))
S23        10   S22 AND IC=G06F
```

18/5/1 (Item 1 from file: 347)
DIALOG(R)File 347:JAPIO

05720282 **Image available**
DEVICE AND **METHOD** FOR **GENERATING** **OPERATING** **SYSTEM**

PUB. NO.: 10-003382 [JP 10003382 A]
PUBLISHED: January 06, 1998 (**19980106)**
INVENTOR(s): IWAMURA YOSHIYUKI
SUMI FUMIO
NISHIHATA MOTOHIDE
APPLICANT(s): MATSUSHITA ELECTRIC IND CO LTD [000582] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 08-156544 [JP 96156544]
FILED: June 18, 1996 (19960618)
INTL CLASS: [6] **G06F-009/06 ; G06F-011/30 ; G06F-011/30 ; G06F-013/00**

JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units);
45.2 (INFORMATION PROCESSING -- Memory Units)

ABSTRACT

PROBLEM TO BE SOLVED: To generate a **small** -sized and high-speed **OS** by generating the **objects** of various functions where an unnecessary processing is removed with an **OS** main source **file** which is edited through the use of a specified **OS** main definition **file** and the **OS** main source **file** as an input.

SOLUTION: An **OS** source editing means 104 edits the **OS** main source **file** with the **OS** main definition **file** 102 and the **OS** main source **file** 101 as the input and a **program** converting means 109 generates the **objects** 112 of **OS** with the edited **OS** main source **file** 105 as the input. Here, the **OS** main definition **file** 102 defines presence or absence in the usage of a system call and a time-out **function** to be used by an **application** program and presence or absence of an error processing. The **OS** main source **file** 101 is adopted as the file where a mark for controlling the erasion or the insertion of a program row through the use of contents which is defined by the **OS** main definition **file** 102 is added.

18/5/2 (Item 2 from file: 347)
DIALOG(R)File 347:JAPIO

04828757 **Image available**
SYSTEM AND DEVICE FOR **GENERATING** PARALLEL **PROGRAM** **SYSTEM**

PUB. NO.: 07-121357 [JP 7121357 A]
PUBLISHED: May 12, 1995 (**19950512)**
INVENTOR(s): YAMANAKA KENJIROU
KOMURA SEIICHI
KATO JUN
ICHIKAWA HARUHISA
APPLICANT(s): NIPPON TELEGR & TELEPH CORP <NTT> [000422] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 05-267120 [JP 93267120]
FILED: October 26, 1993 (19931026)
INTL CLASS: [6] **G06F-009/06**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

ABSTRACT

PURPOSE: To guarantee the connection of a synthesized result and to comprehend parallel **programs** by **generating** **module** **programs** provided with specific operation sequence in a system where plural modules are cooperatively operated.

CONSTITUTION: When a message sequence structured by the three components of selection, continuation and repetition is inputted, a control part 5 stores it in an **operating system** storage part 2 and when the **programs** are not **generated** for the entire **modules** , the part 5 **extracts** the **operating system** of the **module** to be obtained and stores it in one of plural storage parts 7, 8... for storing the operation sequence for the respective modules inside the operation sequence storage part 2. Then, a non-deterministic automation for receiving the **operating system** of the **module** to be obtained is generated, converted to a deterministic automation, optimized and stored in a corresponding **module program** storage part inside a **program** storage part 3. The operation is repeated and when the **programs** are **generated** for the entire **modules** , output to an output part 4 is performed.

18/5/3 (Item 3 from file: 347)
DIALOG(R)File 347:JAPIO

04081760 **Image available**
SIMPLIFICATION SYSTEM FOR **PROGRAM DESIGN**

PUB. NO.: 05-073460 [JP 5073460 A]
PUBLISHED: March 26, 1993 (**19930326)**
INVENTOR(s): SATO HIROCHIKA
APPLICANT(s): NEC COMMUN SYST LTD [491066] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 03-236088 [JP 91236088]
FILED: September 17, 1991 (19910917)
INTL CLASS: [5] **G06F-013/10**
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units)
JOURNAL: Section: P, Section No. 1581, Vol. 17, No. 405, Pg. 129, July 28, 1993 (19930728)

ABSTRACT

PURPOSE: To **simplify** an instruction processing executed to an **OS** /BIOS at the time of designing a software program, and to execute the unitary management with a super-BIOS by constructing a super-BIOS layer between an application layer of a software and an **OS** /BIOS layer.
CONSTITUTION: A processing for generating an instruction of a request to an **OS** /BIOS, executed in an application program is all executed by a super-BIOS 2. The super BIOS executes a multiprocessing by managing concentrically a processing to an **OS** /BIOS 3 requested from each **function** unit of an **application** , and executes management of a hardware apparatus for processing actually an input and an output, etc., through the **OS** /BIOS 3. By constructing the super- BIOS 2, a request processing to a hardware at the time of designing a **program** is **simplified** , and the burden of module management is reduced.

18/5/4 (Item 4 from file: 347)
DIALOG(R)File 347:JAPIO

03546619 **Image available**
PROGRAM GENERATING METHOD

PUB. NO.: 03-209519 [JP 3209519 A]
PUBLISHED: September 12, 1991 (**19910912)**
INVENTOR(s): KAWABATA SHOICHI
APPLICANT(s): SEIKO EPSON CORP [000236] (A Japanese Company or Corporation) , JP (Japan)
APPL. NO.: 02-004192 [JP 904192]
FILED: January 11, 1990 (19900111)
INTL CLASS: [5] **G06F-009/06**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)
JOURNAL: Section: P, Section No. 1285, Vol. 15, No. 483, Pg. 134, December 06, 1991 (19911206)

ABSTRACT

PURPOSE: To obtain an accurate **program** by checking the contents of an environment defining file and **selecting** one of plural files for each of **modules** forming the **program** .

CONSTITUTION: The contents of an environment defining file 31 are checked by an installing program INSTALL 27, and the name of a module file is obtained from a generation table 61. Thus an operator can automatically obtain a desired program 54 without giving any instruction to the program INSTALL 27 and furthermore has no fear to produce the discordance between the environment of an **operating system** and the obtained program 54. Thus the accurate program 54 is obtained.

18/5/8 (Item 3 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

015598936 **Image available**
WPI Acc No: 2003-661091/200362
Related WPI Acc No: 2003-660945; 2003-660949
XRPX Acc No: N03-527375

Run time image generating method, involves combining instances of selected program components **in configuration and creating image by invoking script of each instances in configuration to built component into image**

Patent Assignee: MICROSOFT CORP (MICT)
Inventor: BEACHMAN B J; HILL T J
Number of Countries: 001 Number of Patents: 001
Patent Family:

```
Patent No      Kind   Date     Applicat No     Kind   Date      Week
US 20030121024  A1   20030626  US 2001341511    P   20011213  200362  B
                               US 2002108963    A   20020328
```

Priority Applications (No Type Date): US 2001341511 P 20011213; US 2002108963 A 20020328
Patent Details:

```
Patent No  Kind Lan Pg   Main IPC    Filing Notes
US 20030121024 A1    21 G06F-009/44   Provisional application US 2001341511
```

Abstract (Basic): US 20030121024 A1

NOVELTY - The method involves providing **software program components** with a description data. A **program component** is **selected** , and an instance of each **component** is generated. The instances are combined in a configuration of a run-time image and each instance has script for building a corresponding selected component. The image is created by invoking the script in the configuration to build the component into the run-time image.

DETAILED DESCRIPTION - An INDEPENDENT CLAIM is also included for a system for generating runtime images of a software program.

USE - Used for building run time image of software programs e.g. **operating system** .

ADVANTAGE - The method reduces the complexity by providing a **program** that handles every available **module** of the regular **operating system** , hence capable of **constructing** run-time images from any combination of the modules.

DESCRIPTION OF DRAWING(S) - The drawing shows a schematic view of a configuration object containing scripts that are invoked to build a run-time image.

pp; 21 DwgNo 7/7

Title Terms: RUN; TIME; IMAGE; GENERATE; METHOD; COMBINATION; INSTANCE; SELECT; PROGRAM; COMPONENT; CONFIGURATION; IMAGE; INVOKE; SCRIPT; INSTANCE; CONFIGURATION; BUILD; COMPONENT; IMAGE
Derwent Class: T01
International Patent Class (Main): **G06F-009/44**
File Segment: EPI

```
 18/5/11     (Item 6 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004  Thomson Derwent. All rts. reserv.

015368751     **Image available**
WPI Acc No: 2003-429689/200340
XRPX Acc No: N03-343115
  System for automatic installation and configuration of  operating
  system  uses  identification  data with location and  function  data in
  creating  installation  program  for installing  package  to user
  computer
Patent Assignee: INT BUSINESS MACHINES CORP (IBMC  ); CIE IBM FRANCE (IBMC
  )
Inventor: AGUDO CRESPO A; FERNANDEZ GONZALEZ I; CRESPO A A; GONZALEZ I F
Number of Countries: 101  Number of Patents: 005
Patent Family:
Patent No      Kind   Date    Applicat No    Kind   Date     Week
US 20030046682  A1  20030306  US 2001942134   A   20010829  200340  B
WO 200321430    A2  20030313  WO 2002EP9681   A   20020813  200340
EP 1421483      A2  20040526  EP 2002772234   A   20020813  200435
                              WO 2002EP9681   A   20020813
KR 2004028804   A   20040403  KR 2003717260   A   20031230  200451
AU 2002337038   A1  20030318  AU 2002337038   A   20020813  200452

Priority Applications (No Type Date): US 2001942134 A 20010829
Patent Details:
Patent No  Kind Lan Pg   Main IPC    Filing Notes
US 20030046682 A1    22 G06F-009/445
WO 200321430   A2 E     G06F-009/44
   Designated States (National): AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA
   CH CN CO CR CU CZ DE DK DM DZ EC EE ES FI GB GD GE GH GM HR HU ID IL IN
   IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ
   OM PH PL PT RO RU SD SE SG SI SK SL TJ TM TN TR TT TZ UA UG UZ VN YU ZA
   ZM ZW
   Designated States (Regional): AT BE BG CH CY CZ DE DK EA EE ES FI FR GB
   GH GM GR IE IT KE LS LU MC MW MZ NL OA PT SD SE SK SL SZ TR TZ UG ZM ZW
EP 1421483     A2 E     G06F-009/44   Based on patent WO 200321430
   Designated States (Regional): AL AT BE BG CH CY CZ DE DK EE ES FI FR GB
   GR IE IT LI LT LU LV MC MK NL PT RO SE SI SK TR
KR 2004028804 A         G06F-009/44
AU 2002337038 A1        G06F-009/44   Based on patent WO 200321430

Abstract (Basic): US 20030046682 A1
        NOVELTY - The  system  has a program  generator  that  creates  the
    installation  program  and associated response  files  on a server
    computer based on the identification data to install a package on an
    end-user computer. The identification data comprises of the location
    data where the end-user computer is installed, and function data
    performed by the end-user computer in the network.
        DETAILED DESCRIPTION - The program generator is coupled to a
    sending unit that sends the package from a storage unit to the server
    computer based on the identification data of the end-user computer. The
    storage unit is coupled to the server computer for storing the package
    and identification data of the end-user computer. The package includes
    an  operating   system , software products and applications.
        An INDEPENDENT CLAIM is included for the automatic installation and
    configuration of  operating   system .
        USE - For automatic installation and configuration of an  operating
      system .
        ADVANTAGE - Improves current systems and methods for pristine
    installation of  operating   systems .  Builds   operating   system
    and any required product or application according to machine profiling
    mechanism controlled by central repository. Provides a solution
    integrated with software distribution products to automatically update
    machines after they are first built. Allows tracking of status of
    servers or workstations over their life cycles. Isolates the end-user
    from implementation details such as  operating   system  parameters or
    from product configurations. Portable to other  operating   systems  or
```

databases. Provides central control of deployment. Avoids need for connection to central repository to build machines and need for special skills to install servers or workstations. Assures that base levels of machines are fulfilled. Saves deployment costs and reduces deployment time. Allows machine to be rebuilt any number of times.

DESCRIPTION OF DRAWING(S) - The figure is a flowchart showing the package management process in the automatic installation and configuration of an **operating system** .

pp; 22 DwgNo 2/11

Title Terms: SYSTEM; AUTOMATIC; INSTALLATION; CONFIGURATION; OPERATE; SYSTEM; IDENTIFY; DATA; LOCATE; FUNCTION; DATA; INSTALLATION; PROGRAM; INSTALLATION; PACKAGE; USER; COMPUTER

Derwent Class: T01

International Patent Class (Main): **G06F-009/44 ; G06F-009/445**

File Segment: EPI

18/5/12 (Item 7 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

015368750 **Image available**

WPI Acc No: 2003-429688/200340

XRPX Acc No: N03-343114

Integrated data processing system for managing complete end-to-end software delivery process, has third subsystem for distributing created software **product** package **to target execution units**

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC); CIE IBM FRANCE (IBMC)

Inventor: ARBULU BARTUREN M; DIEZ FERNANDEZ M; FERNANDEZ GONZALEZ I; MARTIN GARIJO E; BARTUREN M A; FERNANDEZ M D; GARIJO E M; GONZALEZ I F

Number of Countries: 101 Number of Patents: 005

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 20030046681	A1	20030306	US 2001943563	A	20010830	200340	B
WO 200321432	A1	20030313	WO 2002EP9683	A	20020813	200340	
EP 1421485	A1	20040526	EP 2002772235	A	20020813	200435	
			WO 2002EP9683	A	20020813		
AU 2002337039	A1	20030318	AU 2002337039	A	20020813	200452	
KR 2004033288	A	20040421	KR 2003717249	A	20031230	200454	

Priority Applications (No Type Date): US 2001943563 A 20010830

Patent Details:

Patent No Kind Lan Pg Main IPC Filing Notes
US 20030046681 A1 15 G06F-009/445
WO 200321432 A1 E G06F-009/445
Designated States (National): AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ OM PH PL PT RO RU SD SE SG SI SK SL TJ TM TN TR TT TZ UA UG UZ VN YU ZA ZM ZW
Designated States (Regional): AT BE BG CH CY CZ DE DK EA EE ES FI FR GB GH GM GR IE IT KE LS LU MC MW MZ NL OA PT SD SE SK SL SZ TR TZ UG ZM ZW
EP 1421485 A1 E G06F-009/445 Based on patent WO 200321432
Designated States (Regional): AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI SK TR
AU 2002337039 A1 G06F-009/445 Based on patent WO 200321432
KR 2004033288 A G06F-017/60

Abstract (Basic): US 20030046681 A1

NOVELTY - The system has a first subsystem for **identifying** within the central repository **software components** of a **software** product to be delivered, a second subsystem for **creating** a **software** product **package** from the **identified software components** , and a third subsystem for distributing the **software** product **package created** by the second subsystem to the target software product execution units.

DETAILED DESCRIPTION - The central repository stores the **software components** of the **software** product.

An INDEPENDENT CLAIM is included for the delivery of software products to target software production execution units in a network environment.

USE - For managing complete end-to-end software delivery process. Used in business supported by a wide array of software applications.

ADVANTAGE - Manages software product along whole life cycle from development to installation in production. Has unique global management system that supports whole software delivery process. Independent of platforms, **operating systems** , repositories or databases utilized in specific implementations. Assures integrity of software systems and accurate performance, tracking and controlling dependencies and relationships between different **software components** and documentation, thus avoiding redundancies and ensuring that all pieces can be plugged together to work as required. Ensures that installable **software packages** are **generated** and distributed independent of package content and of destination systems, and can be installed by unattended procedure. Ensures that new software versions can be built and packaged once and stored in software distribution repository of distribution subsystem, enabling reuse of installation packages in different environments.

DESCRIPTION OF DRAWING(S) - The figure is a flowchart schematically showing the main steps in the integrated data processing system.

pp; 15 DwgNo 3/4

Title Terms: INTEGRATE; DATA; PROCESS; SYSTEM; MANAGE; COMPLETE; END; END; SOFTWARE; DELIVER; PROCESS; THIRD; SUBSYSTEM; DISTRIBUTE; SOFTWARE; PRODUCT; PACKAGE; TARGET; EXECUTE; UNIT
Derwent Class: T01
International Patent Class (Main): **G06F-009/445** ; **G06F-017/60**
International Patent Class (Additional): **G06F-009/44**
File Segment: EPI

18/5/13 (Item 8 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

015250217 **Image available**
WPI Acc No: 2003-311143/200330
XRPX Acc No: N03-247613

Object display system for software **development,** selects software component **as source for graphical representation of** object **created by** application program **, during run time of application program**

Patent Assignee: BROUSSARD S J (BROU-I)
Inventor: BROUSSARD S J
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 20020180792	A1	20021205	US 2001870621	A	20010531	200330	B

Priority Applications (No Type Date): US 2001870621 A 20010531
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 20020180792	A1		35	G06F-003/00	

Abstract (Basic): US 20020180792 A1

NOVELTY - A **software component** is **selected** as source for graphical representation of an **object** created by **application program** written in Java programming language, during run time of the application **program** . The appearance of the displayed **object** is independent of an **operating system** .

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the following:

(1) an object display method; and

(2) a storage device having **object** displaying **program** .

USE - For displaying **object** created by **application program** for **software** development.

ADVANTAGE - Code maintainability is improved greatly. More portable, platform-neutral API is created easily.

DESCRIPTION OF DRAWING(S) - The figure shows a computer system.
pp; 35 DwgNo 1/19

Title Terms: OBJECT; DISPLAY; SYSTEM; SOFTWARE; DEVELOP; SELECT; SOFTWARE; COMPONENT; SOURCE; GRAPHICAL; REPRESENT; OBJECT; APPLY; PROGRAM; RUN; TIME; APPLY; PROGRAM
Derwent Class: T01; T04
International Patent Class (Main): **G06F-003/00**
File Segment: EPI

18/5/15 (Item 10 from file: 350)
DIALOG(R)File 350:Derwent WPIX

014666784 **Image available**
WPI Acc No: 2002-487488/ **200252**
XRPX Acc No: N02-385138

Software environment and user interface for handling software components **, especially bots, includes device for storing properties assigned to each component to be installed**

Patent Assignee: BOTBOX AB (BOTB-N)
Inventor: ERICSSON J; FINNE N; UPPLANDS S J; JANSON S
Number of Countries: 001 Number of Patents: 002
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
SE 200001575	A	20011029	SE 20001575	A	20000428	200252	B
SE 522689	C2	20040224	SE 20001575	A	20000428	200416	

Priority Applications (No Type Date): SE 20001575 A 20000428
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
SE 200001575	A		37	G06F-009/445	
SE 522689	C2			G06F-009/445	

Abstract (Basic): SE 200001575 A

NOVELTY - A device is used to read in properties assigned to **software components** to be installed, e.g. descriptions, codes, states and data from local and/or removable resources. An installation storage device is used to store properties assigned to each **software component** to be installed. As soon as one of the stored **software components** is taken away from the installation storage device, another device (5) is used to remove all the storage properties assigned to the component by the installation storage device. At least one of the electronic devices includes one or more user interaction units, comprising at least one device (1) for offering the user at least part of a local and/or remote resource, or offering the user a reference (e.g. hypertext link) (11) to such a resource, said resource being assigned to at least one **software component** to be installed, the resource (reference) being offered with the aid of display unit (1), a resource presentation (e.g. resource overview) (3) and a user interface view (2). At least one **software component** in the resource presentation can be installed using a first device (11, 12) provided in the user interaction unit in order to allowing installation to be carried out using a drag-and-drop technique performed by the user. A second device provided along with or instead of the first device enables an installation of at least one **software component** in the resource presentation to be carried out by single or double-clicking on a mouse, or by selecting from a pop-up menu, keyboard input or voice instruction. A device (2, 5-8, 9A-9C) is provided for organizing, displaying and/or manipulating installed **software components** according to a common conceptual and interactive method, in the form of files or other entities, which can be nested in folders (8) in a filing system in an **operating system** and its user interface, e.g. Mac **OS** or Microsoft Windows. Also possible is a common conceptual structure with resources in a folder hierarchy having a different presentation and interfaction format, e.g. a 3D graphic presentation and interaction, or a voice generating and instruction system, where e.g. installation, deleting, updates, activation and deactivation are

carried out interactively. A device is provided for saving and resetting the states of active **software components** in between user sessions, these active components being understood to mean components which are continually active or repeatedly activated without necessarily being driven by the user activity.

DETAILED DESCRIPTION - The software environment and user interface are used to handle **software components** , e.g. **software** tools, for activation by devices with a processing capacity, e.g. computers, personal digital assistants (PDAs) or mobile phones. The software environment and user interface are either located inside the device or in a different devices which can be connected in contact with each other. At least one of the devices can be connected in contact with local and/or remote resources comprising files on local storage media, e.g. a hard disk or CD-ROM, and data packets accessible from applications. The remote resources comprise web pages which are read via the Internet, an intranet or via another communication network. An INDEPENDENT CLAIM is also included for a computer-readable program-based product used in conjunction with this software environment.

USE - For Internet or intranet sites offering personal services, information and community (communal/group) software for the end user.

ADVANTAGE - The installation, handling and deletion of **software** components such as bots or other **software** tools is **simplified** . A secure and robust execution and handling **environment** is created for **software components** . The number of user activities required for installing and removing **software components** is minimized. Previously installed **software components** can be completely removed. Security settings can be individually regulated for each **software component** . Previously installed **software components** , the **software** environment and user interface can be all be upgraded using installable **software components** . Cooperation, planning or similar interaction between **software components** is supported, whilst at the same time unexpected or undesirable interaction is avoided. Active **software components** which maintain a link between user sessions are supported.

DESCRIPTION OF DRAWING(S) - Figure 1A shows the user interface displayed on the screen of a monitor for a computer connected to the Internet (drawing includes non-English language text).

Computer monitor (1)
User interface overview (2)
Resource overview associated with web browser (3)
Icons (5-7)
Folder for storing icons (8)
Icon status indicators (9A-9C)
Hypertext link for e.g. chat program (11)
Icon associated with hypertext link (12)
pp; 37 DwgNo 1A/11

Title Terms: SOFTWARE; ENVIRONMENT; USER; INTERFACE; HANDLE; SOFTWARE; COMPONENT; BOT; DEVICE; STORAGE; PROPERTIES; ASSIGN; COMPONENT; INSTALLATION
Derwent Class: T01
International Patent Class (Main): **G06F-009/445**
File Segment: EPI

18/5/16 (Item 11 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014621555 **Image available**
WPI Acc No: 2002-442259/200247
XRPX Acc No: N02-348257

Operating system components identifier **for use with computer, has build structure configured to generate** identified operating system component **loaded table dynamically to maintain updated loaded table of** identified components

Patent Assignee: APPLICA SYSTEMS INC (APPL-N)
Inventor: COLBURN M W; JUSTICE B P; SIEMSEN M B

Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 6367072	B1	20020402	US 9841848	A	19980312	200247	B

Priority Applications (No Type Date): US 9841848 A 19980312
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6367072	B1		11	G06F-009/45	

Abstract (Basic): US 6367072 B1

NOVELTY - An **identifier** structure accesses the component table and interrogates the operating system to determine **operating system components** loaded in table. A build structure coupled to **identifier** structure is configured to **generate identified operating system components** loaded table dynamically in order to maintain an updated loaded table of identified components.

DETAILED DESCRIPTION - An INDEPENDENT CLAIM is included for **operating system components identification** method.

USE - For **identifying operating system component** structure for computer.

ADVANTAGE - The system has ability to **identify** and modify the loaded **components** therefore the modification **programs** can be provided for additional **function** that are not originally provided for the system, thereby providing complete functionality to two users, simultaneously.

DESCRIPTION OF DRAWING(S) - The figure shows a flow diagram of the **operating system** .

pp; 11 DwgNo 5/12

Title Terms: OPERATE; SYSTEM; COMPONENT; IDENTIFY; COMPUTER; BUILD; STRUCTURE; CONFIGURATION; GENERATE; IDENTIFY; OPERATE; SYSTEM; COMPONENT; LOAD; TABLE; DYNAMIC; MAINTAIN; UPDATE; LOAD; TABLE; IDENTIFY; COMPONENT
Derwent Class: T01
International Patent Class (Main): **G06F-009/45**
File Segment: EPI

18/5/20 (Item 15 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

013965694 **Image available**
WPI Acc No: 2001-449908/ **200148**
XRPX Acc No: N01-332960

Computer program product for creating and distributing customized application, builds template comprising modifications to base component loaded in memory and indexes template within storage file

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)
Inventor: BUXTON J J
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 6182279	B1	20010130	US 97910144	A	19970812	200148	B

Priority Applications (No Type Date): US 97910144 A 19970812
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6182279	B1		30	G06F-009/445	

Abstract (Basic): US 6182279 B1

NOVELTY - A storage file (212) is defined. Base components (202) are loaded into the memory. Templates comprising modifications to the base components, are created. The modifications are formatted into template format suitable for storage and the template is indexed within the storage file.

DETAILED DESCRIPTION - The base components are executed. The modifications are made with executing base component itself and are executable by an **operating system** (215) to customize the base

component when executed with the template. The modifications made to the executing base component are examined by a user and the modifications are separated from the base component. INDEPENDENT CLAIMS are also included for the following:

(a) Method for generating template of customizations to base component;

(b) Apparatus for storing templates in object oriented computed system

USE - For creating and distributing customized applications.

ADVANTAGE - Provides a system of intelligent, self contained **software modules** to **construct** larger **application** in a **modular** fashion. Provides a technique to modify **software applications** as desired by an end user in **simplified** and efficient manner.

DESCRIPTION OF DRAWING(S) - The figure shows conceptual view of elements comprising component system.

Base components (202)
Storage file (212)
Operating system (215)
pp; 30 DwgNo 2/9

Title Terms: COMPUTER; PROGRAM; PRODUCT; DISTRIBUTE; APPLY; BUILD; TEMPLATE ; COMPRISE; MODIFIED; BASE; COMPONENT; LOAD; MEMORY; INDEX; TEMPLATE; STORAGE; FILE
Derwent Class: T01
International Patent Class (Main): **G06F-009/445**
File Segment: EPI

18/5/21 (Item 16 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

013907319 **Image available**
WPI Acc No: 2001-391532/ **200142**
XRPX Acc No: N01-288077

System and a method for developing reusable, flexible and platform independent software **using** components **types with an interface and a body followed by automatically generating components based on the developed component types**

Patent Assignee: SIEMENS AG (SIEI)
Inventor: AMLER G; DOHT H; ECKARDT D; HACKL F; KUGLER R; MUELLER M; RENNER K; SCHORRIG H
Number of Countries: 027 Number of Patents: 003
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 1087292	A2	20010328	EP 2000118015	A	20000822	200142	B
JP 2001109628	A	20010420	JP 2000273938	A	20000908	200142	
US 6370682	B1	20020409	US 99397020	A	19990915	200227	
			US 99430311	A	19991029		

Priority Applications (No Type Date): US 99430311 A 19991029; US 99397020 A 19990915
Patent Details:

```
Patent No  Kind Lan Pg   Main IPC    Filing Notes
EP 1087292    A2 E  15 G06F-009/44
   Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT
   LI LT LU LV MC MK NL PT RO SE SI
JP 2001109628 A     15 G06F-009/44
US 6370682    B1       G06F-009/45   Cont of application US 99397020
```

Abstract (Basic): EP 1087292 A2

NOVELTY - The system has **software components** and a coupled **operating system** controlling the **software components** each one having a limited functional scope and processing input data to form new output data. Each component is independent of others and is produced in a reusable **platform** independent fashion. **Application software** is **created** by graphically linking a **subset** of the components.

DETAILED DESCRIPTION - Independent claims describe a system for generating a target platform specific **software** component and a

method of **developing** **software** and a computer program product.

USE - As a system and a method for developing reusable, flexible and platform independent **software** using **components** with an interface and a body.

ADVANTAGE - Ports software across multiple hardware platforms to produce a clearly structured, easily maintained software whose capacity is not affected even at high levels of complexity.

DESCRIPTION OF DRAWING(S) - The drawing shows a data flow chart depicting the inter-connection of several components.

pp; 15 DwgNo 1/6

Title Terms: SYSTEM; METHOD; DEVELOP; REUSE; FLEXIBLE; PLATFORM; INDEPENDENT; SOFTWARE; COMPONENT; TYPE; INTERFACE; BODY; FOLLOW; AUTOMATIC; GENERATE; COMPONENT; BASED; DEVELOP; COMPONENT; TYPE

Derwent Class: T01

International Patent Class (Main): **G06F-009/44** ; **G06F-009/45**

File Segment: EPI

18/5/22 (Item 17 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

013822828 **Image available**

WPI Acc No: 2001-307040/ **200132**

XRPX Acc No: N01-219641

Waveform compiler generation for communication system , **involves** generating **target** code **using behavior model which analyze tasks of created functional model**

Patent Assignee: MOTOROLA INC (MOTI)

Inventor: CHRULSKI C M; MILLER A

Number of Countries: 001 Number of Patents: 001

Patent Family:

```
Patent No     Kind   Date     Applicat No    Kind   Date     Week
US 6175948     B1  20010116  US 9819293      A   19980205  200132  B
```

Priority Applications (No Type Date): US 9819293 A 19980205

Patent Details:

```
Patent No  Kind Lan Pg   Main IPC    Filing Notes
US 6175948    B1    21 G06F-017/50
```

Abstract (Basic): US 6175948 B1

NOVELTY - A functional model of a waveform application is created and is partitioned into tasks. An application model is generated from the functional model. A behavior model of **system** tasks and **application** tasks is **created** using which the tasks are analyzed. Target code is generated using the behavior model and by **selecting** one among the hardware, **software** and **operating** **system** **components** .

DETAILED DESCRIPTION - The reusable **components** include **software** **components** , **operating** **system** **components** , hardware **components** and hardware/ **software** **components** . The **software** **components** include **function** , data structures and sub- **functions** . The hardware/ **software** **components** include interfaces, buffering structures and interface drivers. The functional model is created as a data flow type graph. The behavior model includes state machines and scenarios. The target code is generated by **selecting** at least one of the reusable **components** from a group of **software** **components** , **operating** **system** **components** , hardware **components** and hardware/ **software** **components** . An INDEPENDENT CLAIM is also included for communication system developing method.

USE - In waveform application development for use in communication system, used in development environment for design, verification and maintenance.

ADVANTAGE - Reuse components are employed to facilitate development of communication system hence realizing cost reduction. Offers development environment for design, verification and maintenance.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of waveform compiler methodology.

pp; 21 DwgNo 4/12
Title Terms: WAVEFORM; COMPILE; GENERATE; COMMUNICATE; SYSTEM; GENERATE; TARGET; CODE; MODEL; TASK; FUNCTION; MODEL
Derwent Class: T01
International Patent Class (Main): **G06F-017/50**
File Segment: EPI

18/5/23 (Item 18 from file: 350)
DIALOG(R)File 350:Derwent WPIX

013673866 **Image available**
WPI Acc No: 2001-158078/ **200116**
XRPX Acc No: N01-115088

Memory image file **producing** program **for embedded computers, compiles both application program and source code** operating system modules **to** produce **memory image** file **, that fits into limited memory of computer**

Patent Assignee: ROCKWELL TECHNOLOGIES LLC (ROCW)
Inventor: KEELEY T
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 6138271	A	20001024	US 96669739	A	19960626	200116	B

Priority Applications (No Type Date): US 96669739 A 19960626
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6138271	A		8	G06F-009/45	

Abstract (Basic): US 6138271 A

NOVELTY - A compiler compiles both application program (20') and source code **operating system modules** on the list to produce memory image file. The image file is fit to limited memory of embedded computer. Thus, sufficient **operating system** less than that of full **featured operating system** for **application program** is invoked by standard **operating system** calls.

DETAILED DESCRIPTION - A scanner program receives the application program, scans it for the **operating system** calls to full featured **operating system** (18) to **produce** list of source **code operating system modules** (44). The **modules** are required by the **application program** when run on the embedded computer.

USE - For **designing operating system** for **application** programs used on embedded computers.

ADVANTAGE - Since source code **operating system modules** are **produced** simply, **size** of **operating system** for use with embedded computer is **reduced** .

DESCRIPTION OF DRAWING(S) - The figure shows the schematic representation of the development system for producing **reduced size operating system** , that is embedded with **application program** of computer.

Full **featured operating system** (18)
Application program (20')
Source code **operating system modules** (44)
pp; 8 DwgNo 4/4

Title Terms: MEMORY; IMAGE; FILE; PRODUCE; PROGRAM; EMBED; COMPUTER; COMPILE; APPLY; PROGRAM; SOURCE; CODE; OPERATE; SYSTEM; MODULE; PRODUCE; MEMORY; IMAGE; FILE; FIT; LIMIT; MEMORY; COMPUTER
Derwent Class: T01
International Patent Class (Main): **G06F-009/45**
File Segment: EPI

18/5/27 (Item 22 from file: 350)
DIALOG(R)File 350:Derwent WPIX

```
012285207     **Image available**
WPI Acc No: 1999-091313/ 199908
XRPX Acc No: N99-067259
  Object oriented programming  system  for  developing  stock control
  application  - has main memory for storing various  objects  oriented
  programs  corresponding to various  operating   system
Patent Assignee: IBM CORP (IBMC  ); INT BUSINESS MACHINES CORP (IBMC  )
Inventor: ARNOLD V D; BERG A C; BOHRER K A; BRANE T K A; DAHL T M;
  MICHAELSON T; NILSSON A M; ODEGAARD H; PERNBECK T H O
Number of Countries: 003  Number of Patents: 003
Patent Family:
Patent No     Kind   Date     Applicat No    Kind   Date     Week
JP 10326190    A   19981208  JP 9856321       A   19980309  199908  B
US 5936860     A   19990810  US 97834648      A   19970328  199938
KR 98079731    A   19981125  KR 984664        A   19980216  200004

Priority Applications (No Type Date): US 97834648 A 19970328
Patent Details:
Patent No  Kind Lan Pg   Main IPC    Filing Notes
JP 10326190   A     55 G06F-009/44
KR 98079731   A        G06F-017/60
US 5936860    A        G06F-019/00

Abstract (Basic): JP 10326190 A
        The system (30) includes a main memory (38) in which several
    object  oriented  programs  for performing execution through various
    operating   system  is stored. A control framework data base for
    regulating execution of various programs is stored according to their
    classification. The execution of specific program is performed through
    an user interface (54) connected to CPU (32) based on classification
    result.
        USE - For bank transaction.
        ADVANTAGE - Enables developing control application   program
    quickly. Simplifies  maintenance of  application   program   software
    .
        Dwg.2/34
Title Terms: OBJECT; ORIENT; PROGRAM; SYSTEM; DEVELOP; STOCK; CONTROL;
  APPLY; MAIN; MEMORY; STORAGE; VARIOUS; OBJECT; ORIENT; PROGRAM;
  CORRESPOND; VARIOUS; OPERATE; SYSTEM
Derwent Class: T01
International Patent Class (Main):  G06F-009/44 ;  G06F-017/60 ;
  G06F-019/00
International Patent Class (Additional):  G06F-009/06 ; G06G-007/66
File Segment: EPI


 18/5/28     (Item 23 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004  Thomson Derwent. All rts. reserv.

012228398     **Image available**
WPI Acc No: 1999-034505/ 199903
XRPX Acc No: N99-025849
  Software installation method for specific system hardware - involves
  using script file to record configuration changes resulting from
  downloading software, and modifying  operating   system  according to
  changes in script  file
Patent Assignee: AST RES INC (ASTR-N)
Inventor: CHOYE R J; HARDING H N; KUNG Y L
Number of Countries: 001  Number of Patents: 001
Patent Family:
Patent No     Kind   Date     Applicat No    Kind   Date     Week
US 5842024     A   19981124  US 95397652      A   19950227  199903  B
                             US 97876688      A   19970617

Priority Applications (No Type Date): US 95397652 A 19950227; US 97876688 A
  19970617
Patent Details:
```

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5842024	A		35	G06F-009/445	Cont of application US 95397652

Abstract (Basic): US 5842024 A

The method involves storing a number of **software** **modules** on a mass data storage device (200), for downloading onto a hard disk drive (310). Each **software** **module** comprises a **software** **application** **program** and an associated script **file** , which defines changes to configuration of an **operating** **system** program previously stored on the hard disk drive. The script **file** is **created** by installing the **operating** **system** program onto a second data storage device, saving data stored within the program directories, to temporary directories, installing the software application onto the second data storage device, and effecting requisite changes to the program.

Post-installation data stored within the program directories, is compared with that of the data stored in the temporary directories. Changes to the **operating** **system** **program** resulting from the installation are recorded, and a **software** **module** is **selected** for downloading onto the hard disk drive, of a personal computer in accordance with its hardware **components** . The **selected** **software** **module** is downloaded, and the **operating** **system** is modified according to the configuration changes defined by the script file.

ADVANTAGE - Reduces time required by computer manufacturers to install software onto a computer system's hard disk drive.

Dwg.2/2

Title Terms: SOFTWARE; INSTALLATION; METHOD; SPECIFIC; SYSTEM; HARDWARE; SCRIPT; FILE; RECORD; CONFIGURATION; CHANGE; RESULT; SOFTWARE; MODIFIED; OPERATE; SYSTEM; ACCORD; CHANGE; SCRIPT; FILE

Derwent Class: T01

International Patent Class (Main): **G06F-009/445**

File Segment: EPI

18/5/34 (Item 29 from file: 350)

DIALOG(R)File 350:Derwent WPIX

009941326 **Image available**

WPI Acc No: 1994-209038/ **199425**

XRPX Acc No: N94-164499

Human object oriented object programming system - creating components **each representing element of computer** program **, creating properties associated with each** component **, determining dependencies between** program components **, and storing** components **, associated properties and dependencies**

Patent Assignee: TALIGENT INC (TALI-N)

Inventor: BIANCHI C A; MCINERNEY P J

Number of Countries: 048 Number of Patents: 008

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 5325533	A	19940628	US 9385271	A	19930628	199425	B
WO 9500903	A1	19950105	WO 94US195	A	19940106	199507	
AU 9460219	A	19950117	AU 9460219	A	19940106	199522	
EP 664027	A1	19950726	EP 94906535	A	19940106	199534	
			WO 94US195	A	19940106		
CN 1105507	A	19950719	CN 94190007	A	19940106	199730	
EP 664027	B1	19970723	EP 94906535	A	19940106	199734	
			WO 94US195	A	19940106		
JP 9506722	W	19970630	WO 94US195	A	19940106	199736	
			JP 95502765	A	19940106		
DE 69404439	E	19970904	DE 604439	A	19940106	199741	
			EP 94906535	A	19940106		
			WO 94US195	A	19940106		

Priority Applications (No Type Date): US 9385271 A 19930628

Cited Patents: 05Jnl.Ref; EP 496494

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5325533	A		59	G06F-015/20	

```
WO 9500903    A1    56 G06F-009/44
   Designated States (National): AT AU BB BG BR BY CA CH CN CZ DE DK ES FI
   GB HU JP KP KR KZ LK LU LV MG MN MW NL NO NZ PL PT RO RU SD SE SK UA UZ
   VN
   Designated States (Regional): AT BE CH DE DK ES FR GB GR IE IT LU MC NL
   OA PT SE
AU 9460219    A        G06F-009/44   Based on patent WO 9500903
EP 664027     A1 E  59 G06F-009/44   Based on patent WO 9500903
   Designated States (Regional): CH DE ES FR GB IT LI NL SE
EP 664027     B1 E  48 G06F-009/44   Based on patent WO 9500903
   Designated States (Regional): CH DE ES FR GB LI NL SE
JP 9506722    W    112 G06F-009/06   Based on patent WO 9500903
DE 69404439   E        G06F-009/44   Based on patent EP 664027
                                     Based on patent WO 9500903
CN 1105507    A        G06F-009/44
```

Abstract (Basic): US 5325533 A

The human oriented object programming system provides an interactive and dynamic modelling system to assist in the incremental building of computer programs. A program is modeled as a collection of units called components. A component represents a single compilable language element such as a class or a function. The three major functionality are the database, the compiler and the build mechanism. The database stores the components and properties.

The compiler, along with compiling the source code of a property, is responsible for calculating the dependencies associated with a component. The build mechanism uses properties of components along with the compiler generated dependencies to correctly and efficiently sequence the compilation of components during a build process.

ADVANTAGE - Allows programmer to perform fine granularity source code editing in computer program with optimising incremental compiler in **development** of complex **programs** . Facilitates **development** of complex computer **programs** such as **operating** **systems** and large applications with graphic user interfaces - GUIs.

Dwg.4/25

Title Terms: HUMAN; OBJECT; ORIENT; OBJECT; PROGRAM; SYSTEM; COMPONENT; REPRESENT; ELEMENT; COMPUTER; PROGRAM; PROPERTIES; ASSOCIATE; COMPONENT; DETERMINE; PROGRAM; COMPONENT; STORAGE; COMPONENT; ASSOCIATE; PROPERTIES
Derwent Class: T01
International Patent Class (Main): **G06F-009/06 ; G06F-009/44 ; G06F-015/20**
International Patent Class (Additional): **G06F-009/45**
File Segment: EPI

18/5/35 (Item 30 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

009855910 **Image available**
WPI Acc No: 1994-135766/ **199416**
XRPX Acc No: N94-106608

Automatic development **of** operating system **bootable image - determining if available peripherals include software for basic functioning, and generating relevant boot image, including device drivers, from obtained software**

Patent Assignee: COMPAQ COMPUTER CORP (COPQ)
Inventor: ABMAYR D L; BARRON D L; BURCKHARTT D M; CROSSWY W C; ROSENBLUM H M; ABMAYR D W
Number of Countries: 029 Number of Patents: 003
Patent Family:

```
Patent No     Kind   Date     Applicat No    Kind   Date     Week
WO 9408288     A1   19940414  WO 93US9072     A   19930922  199416  B
US 5325532     A    19940628  US 92951226     A   19920925  199425
AU 9351383     A    19940426  AU 9351383      A   19930922  199432
```

Priority Applications (No Type Date): US 92951226 A 19920925
Cited Patents: EP 429252; EP 448495; GB 2203869; US 5136709

```
Patent Details:
Patent No  Kind Lan Pg   Main IPC    Filing Notes
WO 9408288    A1 E  37 G06F-009/445
   Designated States (National): AT AU BR CA DE DK ES FI GB JP KR NL NO NZ
   PL PT RO RU SE
   Designated States (Regional): AT BE CH DE DK ES FR GB GR IE IT LU MC NL
   OA PT SE
US 5325532    A     14 G06F-009/445
AU 9351383    A        G06F-009/445  Based on patent WO 9408288
```

Abstract (Basic): WO 9408288 A

The automatic **development** of an **operating system** boot image includes detection of available devices. The UNIX (RTM) based computer system includes a variety of devices, a basic ROM boot program and device drivers in known disc positions. On starting the computer, the required **operating system** is determined (232) and an installation program started.

This determines what devices are available to the system and refers to a hard disc to locate related device drivers. The drivers are then combined and a correct boot image written to disc and a restart through this boot initiated.

ADVANTAGE - Allows proper boot image to be created automatically and subsequent changes to be incorporated. Allows transportable **operating system** to be **developed** , which can be readily configured on isolated computer system.

Dwg.7/8

Title Terms: AUTOMATIC; DEVELOP; OPERATE; SYSTEM; IMAGE; DETERMINE; AVAILABLE; PERIPHERAL; SOFTWARE; BASIC; FUNCTION; GENERATE; RELEVANT; BOOT; IMAGE; DEVICE; DRIVE; OBTAIN; SOFTWARE

Derwent Class: T01

International Patent Class (Main): **G06F-009/445**

File Segment: EPI

20/5/1 (Item 1 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

016378903 **Image available**
WPI Acc No: 2004-536810/200452
XRPX Acc No: N04-425315

Built-in software development apparatus for mobile phone, generates table comprising reference address for which symbol reference **is preformed and** connects reference **origination address with** referred **address**

Patent Assignee: MITSUBISHI ELECTRIC CORP (MITQ)
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
JP 2004192604	A	20040708	JP 2003117717	A	20030423	200452	B

Priority Applications (No Type Date): JP 2002300194 A 20021015
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2004192604	A		14	G06F-009/45	

Abstract (Basic): JP 2004192604 A

NOVELTY - A **extraction** unit (122) **extracts** the relation between the **modules** of a **software** using symbol reference. A vector table preparation unit (123) generates a table (115) comprising the reference address for which symbol **reference** is preformed and **connects** the **reference** origination address with the referred address. A preparation unit (124) updates the vector table by including the reference address from other modules.

DETAILED DESCRIPTION - An INDEPENDENT CLAIM is also included for built-in **software development method** .

USE - For **developing software** for information management **system** of auxiliary memory in mobile telephone.

ADVANTAGE - The software program is developed reliably and the problem in the software is solved easily by exchanging the corresponding module.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of the built-in software development apparatus. (Drawing includes non-English language text).

vector table (115)
target program (117)
extraction unit (122)
vector table preparation unit (123)
preparation unit (124)
pp; 14 DwgNo 1/12

Title Terms: BUILD; SOFTWARE; DEVELOP; APPARATUS; MOBILE; TELEPHONE; GENERATE; TABLE; COMPRISE; REFERENCE; ADDRESS; SYMBOL; REFERENCE; PREFORM ; CONNECT; REFERENCE; ADDRESS; REFER; ADDRESS
Derwent Class: T01; W01
International Patent Class (Main): **G06F-009/45**
International Patent Class (Additional): **G06F-009/44**
File Segment: EPI

20/5/5 (Item 5 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014686846 **Image available**
WPI Acc No: 2002-507550/200254
Related WPI Acc No: 2003-352807
XRPX Acc No: N02-401669

Graphic application development system **for magnetic resonance imaging system, has graphical building area in which** selected components **from a library are linked at connection points providing executable** application

Patent Assignee: GE MEDICAL SYSTEMS GLOBAL TECHNOLOGY CO (GENE); DEBBINS J P (DEBB-I); GOULD K L (GOUL-I); LICATO P E (LICA-I); POLZIN J A

```
  (POLZ-I); RADICK M T (RADI-I); SAT G (SATG-I); THOMAS D (THOM-I)
Inventor: DEBBINS J P; GOULD K L; LICATO P E; POLZIN J A; THOMAS D; RADICK
  M T; SAT G
Number of Countries: 024  Number of Patents: 005
Patent Family:
Patent No     Kind   Date    Applicat No    Kind   Date     Week
US 20020060566  A1  20020523  US 2000721233   A   20001122  200254  B
                              US 2001839055   A   20010420
WO 200242987   A2  20020530  WO 2001US46745  A   20011108  200254
KR 2002070345  A   20020905  KR 2002709348   A   20020720  200311
EP 1405176     A2  20040407  EP 2001997775   A   20011108  200425
                             WO 2001US46745  A   20011108
JP 2004514993  W   20040520  WO 2001US46745  A   20011108  200434
                             JP 2002545435   A   20011108

Priority Applications (No Type Date): US 2001839055 A 20010420; US
  2000721233 A 20001122
Patent Details:
Patent No  Kind Lan Pg   Main IPC    Filing Notes
US 20020060566 A1    18 G06F-009/44   CIP of application US 2000721233
WO 200242987  A2 E     G06F-019/00
   Designated States (National): CN IN JP KR
   Designated States (Regional): AT BE CH CY DE DK ES FI FR GB GR IE IT LU
   MC NL PT SE TR
KR 2002070345 A        G06F-019/00
EP 1405176    A2 E     G06F-009/44   Based on patent WO 200242987
   Designated States (Regional): DE NL
JP 2004514993 W     50 G06F-009/44   Based on patent WO 200242987

Abstract (Basic): US 20020060566 A1
        NOVELTY - A user selects and moves components written in JAVA with
    predefined function from a component library to a graphic building
    area. The selected components are graphically linked at connection
     points to provide a software reference to each of the selected
    components representing an executable application , which are
    serialized and downloaded to the magnetic resonance imaging system.
        DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the
    following:
        (1) Graphic application program producing system;
        (2) Graphic application provision off-line programming method.
        USE - Application development system is used for medical
    imaging system e.g. magnetic resonance imaging (MRI) system.
        ADVANTAGE - The graphic building area with multiple icons can be
    easily linked together to make construction and compilation of an
    imaging application easier. Graphic application are developed at
    external, off-line computer without the need of expensive scanning
    equipment.
        DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of
    the magnetic resonance imaging system.
        pp; 18 DwgNo 1/9
Title Terms: GRAPHIC; APPLY; DEVELOP; SYSTEM; MAGNETIC; RESONANCE; IMAGE;
  SYSTEM; GRAPHICAL; BUILD; AREA; SELECT; COMPONENT; LIBRARY; LINK; CONNECT
  ; POINT; EXECUTE; APPLY
Derwent Class: S01; S03; S05; T01
International Patent Class (Main): G06F-009/44 ; G06F-019/00
File Segment: EPI


 20/5/6     (Item 6 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014666784     **Image available**
WPI Acc No: 2002-487488/200252
XRPX Acc No: N02-385138
  Software environment and user interface for handling software components,
  especially bots, includes device for storing properties assigned to each
  component to be installed
```

Patent Assignee: BOTBOX AB (BOTB-N)
Inventor: ERICSSON J; FINNE N; UPPLANDS S J; JANSON S
Number of Countries: 001 Number of Patents: 002
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
SE 200001575	A	20011029	SE 20001575	A	20000428	200252	B
SE 522689	C2	20040224	SE 20001575	A	20000428	200416	

Priority Applications (No Type Date): SE 20001575 A 20000428
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
SE 200001575	A		37	G06F-009/445	
SE 522689	C2			G06F-009/445	

Abstract (Basic): SE 200001575 A

NOVELTY - A device is used to read in properties assigned to software components to be installed, e.g. descriptions, codes, states and data from local and/or removable resources. An installation storage device is used to store properties assigned to each software component to be installed. As soon as one of the stored software components is taken away from the installation storage device, another device (5) is used to remove all the storage properties assigned to the component by the installation storage device. At least one of the electronic devices includes one or more user interaction units, comprising at least one device (1) for offering the user at least part of a local and/or remote resource, or offering the user a **reference** (e.g. hypertext **link**) (11) to such a resource, said resource being assigned to at least one software component to be installed, the resource (reference) being offered with the aid of display unit (1), a resource presentation (e.g. resource overview) (3) and a user interface view (2). At least one software component in the resource presentation can be installed using a first device (11, 12) provided in the user interaction unit in order to allowing installation to be carried out using a drag-and-drop technique performed by the user. A second device provided along with or instead of the first device enables an installation of at least one software component in the resource presentation to be carried out by single or double-clicking on a mouse, or by selecting from a pop-up menu, keyboard input or voice instruction. A device (2, 5-8, 9A-9C) is provided for organizing, displaying and/or manipulating installed software components according to a common conceptual and interactive method, in the form of files or other entities, which can be nested in folders (8) in a filing system in an operating system and its user interface, e.g. Mac OS or Microsoft Windows. Also possible is a common conceptual structure with resources in a folder hierarchy having a different presentation and interfaction format, e.g. a 3D graphic presentation and interaction, or a voice generating and instruction system, where e.g. installation, deleting, updates, activation and deactivation are carried out interactively. A device is provided for saving and resetting the states of active software components in between user sessions, these active components being understood to mean components which are continually active or repeatedly activated without necessarily being driven by the user activity.

DETAILED DESCRIPTION - The software environment and user interface are used to handle software components, e.g. software tools, for activation by devices with a processing capacity, e.g. computers, personal digital assistants (PDAs) or mobile phones. The software environment and user interface are either located inside the device or in a different devices which can be connected in contact with each other. At least one of the devices can be connected in contact with local and/or remote resources comprising files on local storage media, e.g. a hard disk or CD-ROM, and data packets accessible from applications. The remote resources comprise web pages which are read via the Internet, an intranet or via another communication network. An INDEPENDENT CLAIM is also included for a computer-readable program-based product used in conjunction with this software environment.

USE - For Internet or intranet sites offering personal services, information and community (communal/group) software for the end user.

ADVANTAGE - The installation, handling and deletion of software components such as bots or other **software** tools is **simplified** . A secure and robust execution and handling **environment** is **created** for **software** components. The number of user activities required for installing and removing software components is minimized. Previously installed software components can be completely removed. Security settings can be individually regulated for each software component. Previously installed software components, the software environment and user interface can be all be upgraded using installable software components. Cooperation, planning or similar interaction between software components is supported, whilst at the same time unexpected or undesirable interaction is avoided. Active software components which maintain a link between user sessions are supported.

DESCRIPTION OF DRAWING(S) - Figure 1A shows the user interface displayed on the screen of a monitor for a computer connected to the Internet (drawing includes non-English language text).

Computer monitor (1)
User interface overview (2)
Resource overview associated with web browser (3)
Icons (5-7)
Folder for storing icons (8)
Icon status indicators (9A-9C)
Hypertext link for e.g. chat program (11)
Icon **associated** with hypertext **link** (12)
pp; 37 DwgNo 1A/11

Title Terms: SOFTWARE; ENVIRONMENT; USER; INTERFACE; HANDLE; SOFTWARE; COMPONENT; BOT; DEVICE; STORAGE; PROPERTIES; ASSIGN; COMPONENT; INSTALLATION
Derwent Class: T01
International Patent Class (Main): **G06F-009/445**
File Segment: EPI

20/5/7 (Item 7 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014640686 **Image available**
WPI Acc No: 2002-461390/200249

System using application service provider

Patent Assignee: KISSI (KISS-N)
Inventor: KIM H J
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
KR 2002004898	A	20020116	KR 200162399	A	20011010	200249	B

Priority Applications (No Type Date): KR 200162399 A 20011010
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
KR 2002004898	A		1	G06F-017/60	

Abstract (Basic): KR 2002004898 A

NOVELTY - A system using an application service provider is provided so that a production period can be reduced and operation expenses can be cut down by out-sourcing a web-based application.

DETAILED DESCRIPTION - Stock management programs suitable for the respective companies are embodied on a web with **small** initial investment expenses by employing an **application** service provider, thereby reducing time, expenses and manpower. Here, a user, an application service provider server of a lessor and a program are **connected** . The **related** program embodies current works of the company such as the stock, personnel management, wages and attendance management. Accordingly, the initial investment expenses can be remarkably cut down by converting fixed expenses for building the system into variable expenses. That is, the system using the application service provider decreases personnel expenses, excessive hardware expenses and network repair/management expenses, removes

```
    difficulties in usage and education, replaces a high-priced company
    program, provides a hosting service through a data center, manages data
    and creates a supporting operating  system  for the user.
         pp; 1 DwgNo 1/10
Title Terms: SYSTEM; APPLY; SERVICE
Derwent Class: T01
International Patent Class (Main):  G06F-017/60
File Segment: EPI


 20/5/9     (Item 9 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

013514646     **Image available**
WPI Acc No: 2000-686592/200067
XRPX Acc No: N00-507643
  Code downloading method for smart cards used with personal computer
  system, involves forming mapping of referencable item to corresponding
  token, to form data package
Patent Assignee: SUN MICROSYSTEMS INC (SUNM  ); SCHWABE J E (SCHW-I);
  SUSSER J B (SUSS-I)
Inventor: SCHWABE J E; SUSSER J B
Number of Countries: 091  Number of Patents: 009
Patent Family:
Patent No      Kind   Date     Applicat No     Kind   Date     Week
WO 200046667   A2   20000810  WO 2000US2716    A   20000202  200067  B
AU 200035872   A    20000825  AU 200035872     A   20000202  200067
EP 1145107     A2   20011017  EP 2000914488    A   20000202  200169
                              WO 2000US2716    A   20000202
KR 2001093312  A    20011027  KR 2001709728    A   20010802  200223
BR 200007945   A    20020528  BR 20007945      A   20000202  200239
                              WO 2000US2716    A   20000202
CN 1346465     A    20020424  CN 2000805900    A   20000202  200251
JP 2002536744  W    20021029  JP 2000597682    A   20000202  200274
                              WO 2000US2716    A   20000202
US 20030028686  A1   20030206  US 99243108      A   19990202  200313
AU 771699      B2   20040401  AU 200035872     A   20000202  200455

Priority Applications (No Type Date): US 99243108 A 19990202
Patent Details:
Patent No  Kind Lan Pg   Main IPC    Filing Notes
WO 200046667  A2 E  64 G06F-009/00
   Designated States (National): AE AL AM AT AU AZ BA BB BG BR BY CA CH CN
   CR CU CZ DE DK DM EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP
   KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX NO NZ PL PT RO RU SD SE
   SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW
   Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR
   IE IT KE LS LU MC MW NL OA PT SD SE SL SZ TZ UG ZW
AU 200035872  A        G06F-009/00   Based on patent WO 200046667
EP 1145107    A2 E     G06F-009/00   Based on patent WO 200046667
   Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT
   LI LT LU LV MC MK NL PT RO SE SI
KR 2001093312 A        G06F-009/00
BR 200007945  A        G06F-009/00   Based on patent WO 200046667
CN 1346465    A        G06F-009/445
JP 2002536744 W     91 G06F-009/54   Based on patent WO 200046667
US 20030028686 A1        G06F-009/00
AU 771699     B2       G06F-009/00   Previous Publ. patent AU 200035872
                                     Based on patent WO 200046667

Abstract (Basic): WO 200046667 A2
        NOVELTY - The referencable item is mapped to a corresponding token.
    A separate data package is formed using the mapping data. Several token
    types are generated during mapping. The internal token type relates to
    package, class, static field, static method, instance field and virtual
    method. The external token type relates to one of instance field,
    virtual method and interface method.
```

DETAILED DESCRIPTION - The referencable item is described by metadata which relates to package, class, field and method. An export file which includes one metadata and one or more tokens is generated. The data package is produced by including the executable component **referencing** items and export component for **resolving** **references** to specific items. INDEPENDENT CLAIMS are also included for the following:

(a) smart card;
(b) resource constrained device;
(c) method for linking architecture neutral code downloaded to resource constrained computer;
(d) carrier wave for linking architecture neutral code downloaded to resource constrained computer;
(e) computer implemented system for linking architecture neutral code downloaded to resource constrained computer;
(f) **method** for **generating** converted applet (CAP) **file**

USE - For smart cards, resource-constrained personal computer. Also for field programmable array, pagers, cellular phones.

ADVANTAGE - Utilizes resource on a resource limited device effectively using smaller storage space, through unique token identifiers. Enables to **link** and **resolve** **references** to transfer items on resource limited device, thereby **simplifies** data forwarding. The separately developed **applications** can be loaded onto resource limited device and their components can be shared mutually without compromising private secure information.

DESCRIPTION OF DRAWING(S) - The figure illustrates converter operations.

pp; 64 DwgNo 4B/9

Title Terms: CODE; METHOD; SMART; CARD; PERSON; COMPUTER; SYSTEM; FORMING; MAP; ITEM; CORRESPOND; TOKEN; FORM; DATA; PACKAGE
Derwent Class: T01; T04
International Patent Class (Main): **G06F-009/00** ; **G06F-009/445** ; **G06F-009/54**
International Patent Class (Additional): G06K-019/07
File Segment: EPI

```
File 348:EUROPEAN PATENTS 1978-2004/Sep W04
         (c) 2004 European Patent Office
File 349:PCT FULLTEXT 1979-2002/UB=20040930,UT=20040923
         (c) 2004 WIPO/Univentio

Set     Items   Description
S1     137289   OS OR OPERATING()(SYSTEM? ? OR ENVIRONMENT? ?)
S2        165   S1(5N)(REDUC???? OR SHRINK??? OR LOWER??? OR DECREAS??? OR
             TRIM???? OR CONTRACT? OR CONDENS?)(5N)(SIZE OR SCALE OR FEATU-
             RE? ? OR FUNCTION? OR MAGNITUDE)
S3       1431   S1(7N)(SUBSET? ? OR SIMPLIF? OR SMALL??? OR COMPACT? OR MO-
             DULAR? OR FOOTPRINT? ?)
S4        662   S1(5N)(EXTRACT??? OR PARS??? OR SELECT??? OR PICK??? OR ID-
             ENTIF???? OR IDENTIFICATION)(5N)(COMPONENT? ? OR MODULE? ? OR
             FEATURE? ? OR FUNCTION?)
S5     171459   (APPLICATION? ? OR PROGRAM? ? OR CODE OR SOFTWARE OR FILE?
             ? OR OS OR OPERATING()SYSTEM)(3N)(DEVELOP? OR GENERAT? OR CRE-
             AT? OR DESIGN?? OR DESIGNING OR BUILD? OR CONSTRUCT? OR PRODU-
             C? OR AUTHOR??? OR WRIT???)
S6      38621   (APPLICATION? ? OR PROGRAM? ? OR CODE OR SOFTWARE OR FILE?
             ? OR OS OR OPERATING()SYSTEM)(3N)(CREAT??? OR PRODUCTION)
S7        358   S2:S4(50N)S5:S6
S8        268   S7 AND IC=G06F
S9      12452   S1(5N)(COMPONENT? ? OR MODULE? ? OR FEATURE? ? OR FUNCTION?
              ? OR FILE? ? OR LIBRARIES OR LIBRARY OR DLL OR DLLS OR OBJEC-
             T? ? OR API OR APIS)
S10       243   S2:S4(50N)S5:S6(50N)S9
S11       160   (S2 OR S4)(50N)S5:S6(50N)S9
S12       136   S11 AND IC=G06F
S13     35194   S5:S6(3N)(KIT? ? OR TOOLKIT? ? OR PLATFORM? ? OR ENVIRONME-
             NT? ? OR TOOL? ? OR PACKAGE? ? OR SYSTEM? ? OR ARCHITECTURE? ?
              OR SOLUTION? ? OR METHOD? ? OR METHODOLOG??? OR MODULE? ? OR
             FRAMEWORK? ?)
S14       148   S2:S4(50N)S9(50N)S13
S15       127   S14 AND IC=G06F
S16        86   (S2 OR S4)(30N)S9(30N)S13
S17        80   S16 AND IC=G06F
S18        70   S17 AND AC=US/PR
S19        65   S18 AND AY=(1970:2001)/PR
S20        59   S17 AND PY=1970:2001
S21        68   S19:S20
S22    123952   (TRACE? ? OR TRACING OR RESOLV? OR RESOLUTION? ? OR CONNEC-
             T? OR LINK???)(5N)(REFER??? OR REFERENC? OR DEPENDEN? OR ASSO-
             CIAT? OR CORRELAT? OR RELATED)
S23         7   S2:S4(50N)S22(50N)S5:S6
S24       104   S1(50N)S22(50N)S13
S25        94   S23:S24 AND IC=G06F
S26        80   S25 NOT S21
S27        66   S26 AND AC=US/PR
S28        59   S27 AND AY=(1970:2001)/PR
S29        55   S26 AND PY=1970:2001
S30        68   S28:S29
```

21/3,K/4 (Item 4 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

01535037
Method and system for creating and employing an operating system having selected functionality
Verfahren und System zum Erstellen und Verwenden eines Betriebssystems mit ausgewahlter Funktionalitat
Methode et systeme pour creer et employer un systeme d'exploitation ayant une fonctionnalite selectionnee

PATENT ASSIGNEE:
MICROSOFT CORPORATION, (749866), One Microsoft Way, Redmond, WA 98052, (US), (Applicant designated States: all)

INVENTOR:
Burkhardt, Ryan, 22590 NE 99th Way, Redmond, Washington 98053, (US)
Harikrishnan, Seetharaman, 17815 NE 109th Court, Redmond, Washington 98052, (US)
Yaryan, Tom, 3026 36th Avenue SW, Seattle, Washington 98126, (US)
Bond, Richard, 14829 1st Avenue NE, Duvall, Washington 98019, (US)
Shetty, Vijesh, 3849 Klahanie Drive SE Apt. 6-103, Issaquah, Washington 98029, (US)
Jayaseelan, Vijayachandran, 1800 154th Avenue NE, No. E142, Bellevue, Washington 98007, (US)

LEGAL REPRESENTATIVE:
Maiwald Patentanwalts GmbH (101741), Elisenhof, Elisenstrasse 3, 80335 Munchen, (DE)

PATENT (CC, No, Kind, Date): EP 1280058 A2 030129 (Basic)
EP 1280058 A3 040630
APPLICATION (CC, No, Date): EP 2002014081 020701;
PRIORITY (CC, No, Date): US 912864 010724
DESIGNATED STATES: AT; BE; BG; CH; CY; CZ; DE; DK; EE; ES; FI; FR; GB; GR; IE; IT; LI; LU; MC; NL; PT; SE; SK; TR
EXTENDED DESIGNATED STATES: AL; LT; LV; MK; RO; SI
INTERNATIONAL PATENT CLASS: **G06F-009/445**
ABSTRACT WORD COUNT: 119
NOTE:
Figure number on first page: 4

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	200305	1549
SPEC A	(English)	200305	9764
Total word count - document A			11313
Total word count - document B			0
Total word count - documents A + B			11313

INTERNATIONAL PATENT CLASS: **G06F-009/445**

...ABSTRACT A2
A **system** and method of creating and employing an operating system (OS) image having selected functionality. A user, such as an original equipment manufacturer (OEM) of personal computers (PCs) or an information technology professional, selects a set of **OS components** from a plurality of **OS components** available in an installation **OS** image. The **selected** set of **OS components** is installed as the **OS** image on a computer readable medium such as a CD-ROM. The OS image includes a script for interacting with the **OS** image to perform **functions** desired by a user, such as booting a destination computer and installing a reference OS image or recovering the destination...

...SPECIFICATION functions desired by the user.
The invention described below addresses these and other disadvantages.

SUMMARY OF THE INVENTION
The invention **creates** a reduced operating system (OS) image from an installation OS image. The invention also includes the resulting reduced OS image...

...plurality of OS component modifications 408 and corresponding files that relate to the selected set of OS components 404. Identified **OS component** modifications 416 allow selected **OS components** 414 to execute as the reduced OS image 412 on the destination medium 410. The **OS component** modifications 408 replace, supplement, or otherwise modify the plurality of **OS components** 407.

In particular, the modifications 408 to the plurality of **OS components** 407 may include modifying the **OS components** 407 to not write data to non-volatile memory. For example, in the WINDOWS **operating system** environment, the **OS components** of the invention write registry information to volatile memory instead of non-volatile memory. The registry information includes OS configuration...

...CLAIMS A computer readable recovery medium for use with a computer comprising:

an operating system image; and

a text-based script **file** which interacts with the operating system image to direct recovery from failure of software on the computer.

38. The computer...

...from the plurality of operating system components and generating a list of files associated with the selected set of operating **system components** ; and

operating system component modifications which allow the selected set of **operating system components** to execute as the **operating system** image.

48. The system of claim 47, further comprising a text-based script for directing performance of one or more **functions** by the **operating system** image.

49. The system of claim 48, wherein the **functions** include installing an **operating system** on another computer.

50. The system of claim 48, wherein the functions include one or more tasks selected from the...

...application program as a remote thin client for another computer.

51. The system of claim 47, wherein the set of **operating** system components is a subset of the plurality of operating system components.

52. The system of claim 47, further comprising...

...or more processors to perform acts including allowing the user to add at least one of the additional operating system **components** to the selected set of **operating system components .**

53. The system of claim 47, wherein the **operating system component** modifications include **writing** state information to volatile memory.

54. The system of claim 47, wherein the application program, when executed, further causes the one or more processors to perform acts including **generating** the **operating system** image.

55. The **system** of claim 47, wherein the application program, when executed, further causes the one or more processors to perform acts including...

21/3,K/7 (Item 7 from file: 348)

DIALOG(R)File 348:EUROPEAN PATENTS

(c) 2004 European Patent Office. All rts. reserv.

00747354

Capability engine method and apparatus for a microkernel data processing system

Verfahren und Gerat mit Fahigkeitsvorrichtung fur ein Mikrokern-Datenverarbeitungssystem

Methode et appareil a dispositif de capacite pour un systeme de traitement de donnees a micro-noyaux

PATENT ASSIGNEE:

International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (Proprietor designated states: all)

INVENTOR:
Magee, James Michael, 5310 Canal Drive, Lake Worth, Florida 33463, (US)
Sotomayor, Guy Gil, Jr., 6042 Sherwood Glen Way, Apt. 3, West Palm Beach, Florida 33415, (US)
Youngworth, Christopher Dean, 3 Gulfview Court, Savoy, Illinois 61874, (US)
LEGAL REPRESENTATIVE:
Williams, Julian David (75461), IBM United Kingdom Limited, Intellectual Property Department, Hursley Park, Winchester, Hampshire SO21 2JN, (GB)

```
PATENT (CC, No, Kind, Date):  EP 704796  A2  960403 (Basic)
                              EP 704796  A3  980701
                              EP 704796  B1  000419
APPLICATION (CC, No, Date):   EP 95304188 950616;
PRIORITY (CC, No, Date): US 263313 940928
```

DESIGNATED STATES: AT; BE; CH; DE; ES; FR; GB; IT; LI; NL; SE
INTERNATIONAL PATENT CLASS: **G06F-009/46**
ABSTRACT WORD COUNT: 141
NOTE:
Figure number on first page: 1

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	200016	1721
CLAIMS B	(German)	200016	1710
CLAIMS B	(French)	200016	1919
SPEC B	(English)	200016	37863
Total word count - document A			0
Total word count - document B			43213
Total word count - documents A + B			43213

INTERNATIONAL PATENT CLASS: **G06F-009/46**

...SPECIFICATION MACH project was conducted at the Carnegie Mellon University in the 1980's. The goal of that research was to **develop** a new operating system that would allow computer programmers to exploit modern hardware architectures emerging and yet reduce the size and the number of **features** in the kernel **operating system** . The kernel is the part of an **operating system** that performs basic **functions** such as allocating hardware resources. In the case of the MACH kernel, five programming abstractions were established as the basic...

...top of which the typical complex operations could be built externally to the kernel. The Carnegie Mellon MACH kernel was **reduced in size** in its release 3.0, and is a fully **functional operating system** called the MACH microkernel. The MACH microkernel has the following primitives: the task, the thread, the port, the message, and...

21/3,K/17 (Item 17 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00499055
COMPUTER-AIDED SOFTWARE ENGINEERING FACILITY
RECHNERUNTERSTUTZTE SOFTWAREENTWICKLUNGSEINRICHTUNG
UNITE D'INGENIERIE DU LOGICIEL ASSISTEE PAR ORDINATEUR
PATENT ASSIGNEE:
SEER TECHNOLOGIES, INC., (1382081), 8000 Regency Parkway, Cary, North Carolina 27511, (US), (applicant designated states: AT;BE;CH;DE;DK;ES;FR;GB;IT;LI;LU;NL;SE)
INVENTOR:
WADHWA, Vivek, K., 474 Lincoln Drive, Paramus, NJ 07652, (US)
ATAIE, Faraz, 492 11th Street 4D, Brooklyn, NY 11215, (US)
AUBRUN, Vincent, P., 315 West End Avenue Apt. 7A, New York, NY 10023, (US)
ERLIKH, Leonide, 2819 West 12th Street 16P,, Brooklyn, NY 11224, (US)
FISCHER, Michael, 240 Park Avenue, Passaic, NJ 07055, (US)
FOCHLER, Michael, 301 Cathedral Parkway Apt. 14G, New York, NY 10026,

(US)
HAYMAN, Craig, B., 377 Rector Place 22E, New York, NY 10280, (US)
HILDEBRAND, Daniel, 38 Ranson Street Apt. 2B, Stamford, CT 06902, (US)
HUGHES, James, 36 Wilson Street, Hartsdale, NY 10530, (US)
LAMBERT, Jeffrey, L., 8 Tremblay Road, East Brunswick, NJ 08816, (US)
LEE, Douglas, E., 10 Lake Street Apt. 3D, White Plains, NY 10603, (US)
LIM, Nicholas, R., 201 C Tufnell Park Rd., London N7 0PU, (GB)
MODI, Rajan, S., 372 CPW 48, New York, NY 10025, (US)
MOSEBACH, Richard, W., 100 Sixth Street, Hicksville, NY 11801, (US)
MOSKOWITZ, Joel, M., 480 East 74th Street Apt. 3B, New York, NY 10021, (US)
OLOWU, Tayo, 630 First Avenue Apt. 23B, New York, NY 10016, (US)
POWER, Elaine, C., 54 St. Marks Place, Apt. 2, New York, NY 10003, (US)
SHING, Norman, 710 New Hyde Park Road, New Hyde Park, NY 11040, (US)
LEGAL REPRESENTATIVE:
Goodman, Christopher (31122), Eric Potter & Clarkson St. Mary's Court St. Mary's Gate, Nottingham NG1 1LE, (GB)
PATENT (CC, No, Kind, Date): EP 502975 A1 920916 (Basic)
EP 502975 A1 930331
EP 502975 B1 970716
WO 9108543 910613
APPLICATION (CC, No, Date): EP 91901023 901130; WO 90US7013 901130
PRIORITY (CC, No, Date): US 444060 891130
DESIGNATED STATES: AT; BE; CH; DE; DK; ES; FR; GB; IT; LI; LU; NL; SE
INTERNATIONAL PATENT CLASS: **G06F-017/50** ; **G06F-019/00** ; **G06F-009/45** ; **G06F-009/44**
NOTE:
No A-document published by EPO
LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	EPAB97	2661
CLAIMS B	(German)	EPAB97	2283
CLAIMS B	(French)	EPAB97	2953
SPEC B	(English)	EPAB97	17834

Total word count - document A 0
Total word count - document B 25731
Total word count - documents A + B 25731

INTERNATIONAL PATENT CLASS: **G06F-017/50** ...

... **G06F-019/00** ...

... **G06F-009/45** ...

... **G06F-009/44**

...CLAIMS input which relates the logic constructs to a process entity in the storage area;

q. the data processor operates to **generate** computer **program** **modules** by utilizing the logic constructs, the corresponding pre-selected ones of the process entities, other pre-selected ones of the...

...connectivity data of the hardware components linked to the pre-selected corresponding ones of the process entities, said computer program **modules** each comprising computer code that is supported by the **operating** **environment** of the hardware **component** linked to the corresponding pre- **selected** one of the process entities.

2. The method of claim 1 further characterized in that :
a. the data processor operates to scan the storage area to associate with each **generated** **code** **module** the linked ones of the hardware components; and
b. the data processor operates to distribute each **generated** **code** **module** to the linked ones of the hardware components.
3. The method of claim 1 further characterized in that: each computer...

...further characterized in that the source code comprises statements in a third-generation programming language that is supported by the **operating** environment of a pre-selected hardware component.

9. The method of claim 7 further characterized in that the source code... module.

12. The method of claim 3 further characterized in that :

a. a data processor operates to compile each said **generated** source **code module** and **create** a corresponding object **code module** that is executable in the **operating environment** of the pre-**selected** ones of the corresponding hardware **components** ; and

b. a data processor operates to link each **generated** object **code module** to pre-selected other ones of the object code modules, according to the process relationships specified by the hierarchical process...

21/3,K/19 (Item 19 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00324409
Software management structure
Software-Verwaltungsstruktur
Structure pour la gestion de logiciels
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: BE;CH;DE;ES;FR;GB;IT;LI;NL;SE)
INVENTOR:
Calvert, Nathaniel, 3102 Crescent Lane N.W., Rochester Minnesota 55901, (US)
Effle, James Scott, 4024 2nd Street N.W., Rochester Minnesota 55901, (US)
Johnston, David Lowry, 110 Linden Court S.W., Rochester Minnesota 55902, (US)
Naylor, James Lee, 5718 25th Avenue N.W., Rochester Minnesota 55901, (US)
Olsen-Williams, Helen Marie, 913 8th Avenue S.W., Rochester Minnesota 55902, (US)
Satin, Robert Harry, 607 19th Street N.W. Apt. 24, Rochester Minnesota, (US)
Shaffer, Dennis Lee, 2614 14th Avenue N.W., Rochester Minnesota 55901, (US)
Turk, Gary Albert, 1600 5th Street S.W., Rochester Minnesota 55902, (US)
LEGAL REPRESENTATIVE:
de Pena, Alain et al (15151), Compagnie IBM France Departement de Propriete Intellectuelle, 06610 La Gaude, (FR)

```
PATENT (CC, No, Kind, Date):  EP 317477  A2  890524 (Basic)
                              EP 317477  A3  911204
                              EP 317477  B1  960814
APPLICATION (CC, No, Date):   EP 88480057 881011;
PRIORITY (CC, No, Date): US 122293 871118
```

DESIGNATED STATES: BE; CH; DE; ES; FR; GB; IT; LI; NL; SE
INTERNATIONAL PATENT CLASS: **G06F-009/44**
ABSTRACT WORD COUNT: 135

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

```
Available Text  Language   Update    Word Count
      CLAIMS A  (English)  EPABF1      618
      CLAIMS B  (English)  EPAB96      943
      CLAIMS B   (German)  EPAB96      944
      CLAIMS B   (French)  EPAB96     1066
      SPEC A    (English)  EPABF1    10875
      SPEC B    (English)  EPAB96    10977
Total word count - document A        11494
Total word count - document B        13930
Total word count - documents A + B   25424
```

INTERNATIONAL PATENT CLASS: **G06F-009/44**

...SPECIFICATION words secondary functions, but does not need any of the other secondary functions. Likewise, some of the primary and secondary **functions** of accounting program 33 and **operating system** 34 have been **selected** , and some have been omitted based on the needs of user 41. Application developer 26 repackages programs 31, 33, and...

...computer 51. Application packages 62-64 are created in a similar manner based on the needs of users 42-44. **Application developer** 26 uses packaging **tools** 90 to **create application packages** 61-64, as will be discussed later.
Figure 3 shows application package 61 created by application developer 26 for user...

...SPECIFICATION words secondary functions, but does not need any of the other secondary functions. Likewise, some of the primary and secondary **functions** of accounting program 33 and **operating system** 34 have been **selected** , and some have been omitted based on the needs of user 41. Application developer 26 repackages programs 31, 33, and...

...computer 51. Application packages 62-64 are created in a similar manner based on the needs of users 42-44. **Application developer** 26 uses packaging **tools** 90 to **create application packages** 61-64, as will be discussed later.
Figure 3 shows application package 61 created by application developer 26 for user...

21/3,K/32 (Item 9 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT
(c) 2004 WIPO/Univentio. All rts. reserv.

00799787 **Image available**
ARCHITECTURES FOR NETCENTRIC COMPUTING SYSTEMS
ARCHITECTURES DESTINEES A DES SYSTEMES INFORMATIQUES S'ARTICULANT AUTOUR D'INTERNET
Patent Applicant/Assignee:
ACCENTURE LLP, 100 South Wacker Drive, Chicago, IL 60603, US, US (Residence), US (Nationality)
Inventor(s):
GOODMAN Marina, 6540 W. Irving Park, Chicago, IL 60634, US,
MESOY Tor, Storengv. 63A, N-1368 Stabekk, NO,
TAYLOR Stanton J, 31475 N. Reigate, Green Oaks, IL 60048, US,
REITER Scott R, 504 W. Belden Avenue, Chicago, IL 60614, US,
BOWEN Michael T, Apt. 136, 11600 Audelia Road, Dallas, TX 75243, US,
SIGMUND Larry, 443 Sunset Dr., Crystal Lake, IL 60014, US,
AURIEMMA Ralph, 7242 Pensacola Avenue, Norridge, IL 60706, US,
ALAIRYS Tamara D, 122 N. Charlotte, Lombard, IL 60148, US,
DEGIORGIO Chris M, 860 W. Buckingham, #2W, Chicago, IL 60657, US,
JOHNSON Lizbeth Coleman, 3155 Palm Tree Drive, Lithonia, GA 30038, US,
Legal Representative:
MCCONNELL Dean E (agent), Brinks Hofer Gilson & Lione, One Indiana Square, Suite 2425, Indianapolis, IN 46204, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200133349 A2-A3 **20010510** (WO 0133349)
Application: WO 2000US30519 20001103 (PCT/WO US0030519)
Priority Application: US 99163477 19991103; US 2000676227 20000929
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CR CU CZ DE DK DM DZ EE
ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT
LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM
TR TT TZ UA UG UZ VN YU ZA ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
(OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English

Filing Language: English
Fulltext Word Count: 87917

Patent and Priority Information (Country, Number, Date):
  Patent: ... **20010510**
Main International Patent Class: **G06F-009/44**
International Patent Class: **G06F-009/46**
Fulltext Availability:
  Detailed Description
Publication Year: **2001**

Detailed Description
... architecture 18 can be described as the things a person can see. The platform architecture 18 provides a framework for **selecting** the platform **components** required.

servers, workstations, **operating systems** and networks. The platform architecture 18 represents the overall technology platform for the implementation and deployment of the previously mentioned...

...architecture 36 and the software applications that are included in these architectures 329 34, 36.

Referring to FIG. 3, the **development architecture** 34 includes a common user interface 40, at least one process management tool 42, at least one personal productivity tool...

21/3,K/50 (Item 27 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00731929 **Image available**
PRE-DEFINED HARDWARE AND SOFTWARE BUNDLE READY FOR DATABASE APPLICATIONS
ENSEMBLE MATERIEL LOGICIEL PREDEFINI PRET POUR APPLICATIONS DE TYPE BASE DE DONNEES
Patent Applicant/Assignee:
  ORACLE CORPORATION, 500 Oracle Parkway, Redwood Shores, CA 94065, US, US (Residence), US (Nationality)
Inventor(s):
  ELLISON Lawrence J, 500 Oracle Parkway, Redwood Shores, CA 94065, US
  DOHERTY C Gregory, 36 Downey Street, San Francisco, CA 94117, US
  ROSSITER Jay, 164 Lois Lane, Palo Alto, CA 94303, US
  STOWELL David, 18253 Chelmsford Drive, Cupertino, CA 95014, US
  NAKOS Dimitris, 2220 Middlefield Road, Palo Alto, CA 94301, US
  BHADURI Pari, 1043 A Foster City Boulevard, Foster City, CA 94403, US
  MALLA Venkat, 6133 Linford Terrace, Fremont, CA 94555, US
  CHEN Jane, 395 Richmond Drive #18, Millbrae, CA 94030, US
  KUMARESAN Bala, 625 Old Country Road #215, Belmont, CA 94002, US
  JOHN E So John, 1235 Jefferson Avenue #102, Redwood City, CA 94062, US
  BANERJEE Niloy, 2675 Fayette Drive #207, Mountain View, CA 94040, US
  MATURI Srinivas, 471 Hillsdale Avenue, Santa Clara, CA 95051, US
  GANTI Lalitha, 639 Old County Road #13, Belmont, CA 94402, US
  VENKATARAMAN Ramkumar, 1208 Admiralty Lane, Foster City, CA 94404, US
  STHANIKAM Balasubramanyam, 1107 Second Avenue #501, Redwood City, CA 94063, US
  OU Yuanjiang, 1230 Whipple Avenue #307, Redwood City, CA 94062, US
  KOONEN Joseph Therrattil, #754 9 Cross, Indira Nagar II Stage, Bangalore 560 038, Karnataka, IN
  RAWAT Harish, House No. 016, Mukund Apartments, Palmgrove Road, Bangalore, Karnataka 56004-7, IN
  OLAGAPPAN Muthu, No. 9, Jeevaanandam Street, Vinayagapuram, Ambattur, Chennai, TamiNadu, IN
  SIVAKUMAR Thivakaram Prakash, 7th floor Sona Towers, 71 Miller Road, Bangalore, 506 052, IN
Legal Representative:
  HICKMAN Brian D, McDermott, Will & Emery, 600 13th Street, N.W., Washington, DC 20005-3096, US

Patent and Priority Information (Country, Number, Date):
  Patent: WO 200045259 A1 **20000803** (WO 0045259)
  Application: WO 2000US1995 20000128 (PCT/WO US0001995)
  Priority Application: US 99240229 19990129
Designated States:
(Protection type is "patent" unless otherwise stated - for applications
prior to 2004)
  AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CZ DE DK DM EE ES FI GB GD
  GE GH GM HR HU ID IL IN IS JP KE KG KR KZ LC LK LR LS LT LU LV MA MD MG
  MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ
  VN YU ZA ZW
  (EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE
  (OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
  (AP) GH GM KE LS MW SD SL SZ TZ UG ZW
  (EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 11423

Patent and Priority Information (Country, Number, Date):
  Patent: ... **20000803**
Main International Patent Class: **G06F-009/445**
International Patent Class: **G06F-017/30**
Fulltext Availability:
  Detailed Description
Publication Year: **2000**

Detailed Description
... specifically tailored to execute database server 1 12. In general, the generation of operating system 1 14 may involve initially **writing** a special purpose operating system based on the needs of database server 112, or modifying an existing general purpose operating system to create operating system 1 14. In either case, a first step in the **generation** of **operating system** 114 is determining the services needed by database server 112.

According to one embodiment, the source code of database server...

...system functions. The operating system calls thus identified are considered calls to " required services". After the required services are identified, **operating system** 114 is **generated** to provide specific support of those required services.

In an embodiment that **generates operating system** 1 14 by modifying an
5 existing general purpose operating system, operating system 1 14 may be **generated** by (1) removing **operating system** code that corresponds to services that are not required services, (2) tuning the parameters and configuration of required services to...

21/3,K/51   (Item 28 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT
(c) 2004 WIPO/Univentio. All rts. reserv.

00730893   **Image available**
A SYSTEM AND METHOD FOR TESTING AND VALIDATING DEVICES HAVING AN EMBEDDED OPERATING SYSTEM
SYSTEME ET PROCEDE D'ESSAI ET DE VALIDATION DE DISPOSITIFS DOTES D'UN SYSTEME D'EXPLOITATION INTEGRE
Patent Applicant/Assignee:
  BSQUARE CORPORATION, 3150 139th Avenue S.E., Suite 500, Bellevue, WA 98005-4081, US, US (Residence), -- (Nationality)
Inventor(s):
  GREGORY Peter R, 600 84th Avenue N.E., Medina, WA 98039, US
  WALTERS James Floyd, 1101 Seneca Street #404, Seattle, WA 98101, US
  DIKKALA Janardhana Rao, 1590 134th Avenue S.E. #C208, Bellevue, WA 98005, US

SAMPLE Ian, 5256 University Way N.E. #101, Seattle, WA 98105, US
Legal Representative:
FLORES Victor, LaRiviere, Grubman & Payne, LLP, P.O. Box 3140, Monterey, CA 93942-3140, US
Patent and Priority Information (Country, Number, Date):
Patent: WO 200043880 A1 **20000727** (WO 0043880)
Application: WO 2000US1583 20000121 (PCT/WO US0001583)
Priority Application: US 99116824 19990121; US 99137629 19990604
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM EE ES FI GB
GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA
MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA
UG UZ VN YU ZA ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE
(OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
(AP) GH GM KE LS MW SD SL SZ TZ UG ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 8979

Patent and Priority Information (Country, Number, Date):
Patent: ... **20000727**
Main International Patent Class: **G06F-011/00**
Fulltext Availability:
Claims
Publication Year: **2000**
Claim
... in Claim 5, wherein:
said system stress-testing routine comprise a code base for stress testing said at least one **component** of said **operating system** , said at least one **component** of said **operating system** being **selected** from a group of **operating system components** comprising an Ethernet/NDIS, PCMIA, a memory, a file system, a serial port a video system having a plurality of...

...touch panel, a mouse, a keyboard, and an audio/wave system, said test identifying at least three defects, namely, hardware **design** , hardware programming, and **operating system** interaction, and being executed in automatic or manual mode.

7 The computer-based system, as recited in Claim 1, wherein...

...suites in the form of system stress-testing routine comprising a code base for stress testing said at least one **component** of said **operating system** , said at least one **component** of said **operating system** being **selected** from a group of **operating system components** comprising an Ethernet/NDIS, PCMIA, a memory, a file system, a serial port a video system having a plurality of...
...touch panel, a mouse, a keyboard, and an audio/wave system, said test identifying at least three defects, namely, hardware **design** , hardware programming, and **operating system** interaction, and being executed in automatic or manual mode.

18 A computer-based method for testing and validating an embedded...

21/3,K/53 (Item 30 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00518042 **Image available**
APPLICATION PROGRAM INTERFACES IN AN OPERATING SYSTEM
INTERFACES DE PROGRAMME D'APPLICATION DANS UN SYSTEME D'EXPLOITATION
Patent Applicant/Assignee:
MICROSOFT CORPORATION,

```
  GAGNE Rejean,
  CAJOLET Claude,
Inventor(s):
  GAGNE Rejean,
  CAJOLET Claude,
Patent and Priority Information (Country, Number, Date):
  Patent:               WO 9949394 A1  19990930
  Application:          WO 99US6223 19990322  (PCT/WO US9906223)
  Priority Application: US 9878946 19980323
Designated States:
(Protection type is "patent" unless otherwise stated - for applications
prior to 2004)
  AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CU CZ DE DK EE ES FI GB GD GE
  GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD MG MK
  MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT UA UG UZ VN YU
  ZA ZW GH GM KE LS MW SD SL SZ UG ZW AM AZ BY KG KZ MD RU TJ TM AT BE CH
  CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE BF BJ CF CG CI CM GA GN GW
  ML MR NE SN TD TG
Publication Language: English
Fulltext Word Count: 50023

Patent and Priority Information (Country, Number, Date):
  Patent:               ... 19990930
Main International Patent Class:  G06F-009/46
Fulltext Availability:
  Detailed Description
Publication Year:  1999

Detailed Description
...  is modular. This allows an embedded system designer to
  create an operating environment that is optimized for their unique
  hardware
   development  platform  and  application . The  developer  can  select
  varying
  combinations of the above-described  modules  and  components  for
  inclusion in the  operating  environment . For example, a developer can
  build  an embedded  operating  system  that contains the kernel and a
  selected set of communications but does not provide a graphical user
  interface. Thus, the...


 21/3,K/66     (Item 43 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT
(c) 2004 WIPO/Univentio. All rts. reserv.

00260123
AUTOMATIC DEVELOPMENT OF OPERATING SYSTEM BOOT IMAGE
DEVELOPPEMENT AUTOMATIQUE D'UNE IMAGE AMORCABLE D'UN SYSTEME D'EXPLOITATION
Patent Applicant/Assignee:
  COMPAQ COMPUTER CORPORATION,
Inventor(s):
  CROSSWY William C,
  BARRON Dwight L,
  ABMAYR David L,
  ROSENBLUM Harvey M,
  BURCKHARTT David M,
Patent and Priority Information (Country, Number, Date):
  Patent:               WO 9408288 A1  19940414
  Application:          WO 93US9072 19930922  (PCT/WO US9309072)
  Priority Application: US 92226 19920925
Designated States:
(Protection type is "patent" unless otherwise stated - for applications
prior to 2004)
  AT AU BR CA DE DK ES FI GB JP KR NL NO NZ PL PT RO RU SE AT BE CH DE DK
  ES FR GB GR IE IT LU MC NL PT SE BF BJ CF CG CI CM GA GN ML MR NE SN TD
  TG
Publication Language: English
Fulltext Word Count: 6715
```

Patent and Priority Information (Country, Number, Date):
Patent: ... **19940414**
Main International Patent Class: **G06F-009/445**
Fulltext Availability:
Detailed Description

English Abstract
...removable medium such as a floppy disk or CD-ROM is present a configuration mode in entered when final driver **files** and **operating system modules** are stored on a **selected** hard disk. After this storage the device driver **modules** and **operating system modules**. necessary to develop a boot image of the **operating system** are gathered and linked. The boot image is generated and stored, allowing use on the following boot operations. The computer...
Publication Year: **1994**

Detailed Description
... the system ROM, A list of the system board identification and the identification of all the various adapters in the **system** is then developed, A particular **program** , referred to as INSTALL, for the particular OS is then initiated, The purpose of the INSTALL program is to gather all the various adaptor and **operating systems files** according to the adapter **identification** , the system board **identification** and **operating system** preference, After all the various **modules** , both adapter and **operating system** related, have been gathered,, in the caseof a UNIX environment, a linker and loader is executed to combine and link...

21/3,K/67 (Item 44 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT
(c) 2004 WIPO/Univentio. All rts. reserv.

00251662
FILE DIRECTORY STRUCTURE GENERATOR AND RETRIEVAL TOOL
GENERATEUR DE STRUCTURE POUR REPERTOIRE DE FICHIERS, ET OUTIL D'EXTRACTION
Patent Applicant/Assignee:
2010 SOFTWARE CORPORATION,
Inventor(s):
COHEN-LEVY Leon,
Patent and Priority Information (Country, Number, Date):
Patent: WO 9325961 A1 **19931223**
Application: WO 93US5556 19930610 (PCT/WO US9305556)
Priority Application: US 92896514 19920610
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
CA JP AT BE CH DE DK ES FR GB GR IE IT LU MC NL PT SE
Publication Language: English
Fulltext Word Count: 16722

Patent and Priority Information (Country, Number, Date):
Patent: ... **19931223**
Main International Patent Class: **G06F-009/44**
International Patent Class: **G06F-15:40**
Fulltext Availability:
Detailed Description
Publication Year: **1993**

Detailed Description
... now to Fig. 10b, there is illustrated the readAndDisplayDirectory module flow chart called from step 205 of Fig. 10a. The **module** is to display the **selected files** in an **operating system** directory, as well as the related information from the File Base 14 that was

stored in each file's 11...

...chart is self explanatory. In step 216, the file is examined to see if there was a related 11.DEF11 **file created** by the save **module** 12, If so, and there is an associated 11.DEF11 file containing information about the file stored, a Directory Entry...

```
File 275:Gale Group Computer DB(TM) 1983-2004/Oct 07
         (c) 2004 The Gale Group
File 621:Gale Group New Prod.Annou.(R) 1985-2004/Oct 07
         (c) 2004 The Gale Group
File 636:Gale Group Newsletter DB(TM) 1987-2004/Oct 07
         (c) 2004 The Gale Group
File  16:Gale Group PROMT(R) 1990-2004/Oct 07
         (c) 2004 The Gale Group
File 160:Gale Group PROMT(R) 1972-1989
         (c) 1999 The Gale Group
File 148:Gale Group Trade & Industry DB 1976-2004/Oct 07
         (c)2004 The Gale Group
File 624:McGraw-Hill Publications 1985-2004/Sep 20
         (c) 2004 McGraw-Hill Co. Inc
File  15:ABI/Inform(R) 1971-2004/Oct 06
         (c) 2004 ProQuest Info&Learning
File 647:CMP  Computer Fulltext 1988-2004/Sep W4
         (c) 2004 CMP Media, LLC
File 674:Computer News Fulltext 1989-2004/Sep W1
         (c) 2004 IDG Communications
File 696:DIALOG Telecom. Newsletters 1995-2004/Oct 05
         (c) 2004 The Dialog Corp.
File 369:New Scientist 1994-2004/Sep W4
         (c) 2004 Reed Business Information Ltd.

Set     Items   Description
S1    1004888   OS OR OPERATING()(SYSTEM? ? OR ENVIRONMENT? ?)
S2       1010   S1(5N)(REDUC???? OR SHRINK??? OR LOWER??? OR DECREAS??? OR
             TRIM???? OR CONTRACT? OR CONDENS?)(5N)(SIZE OR SCALE OR FEATU-
             RE? ? OR FUNCTION? OR MAGNITUDE)
S3      19539   S1(7N)(SUBSET? ? OR SIMPLIF? OR SMALL??? OR COMPACT? OR MO-
             DULAR? OR FOOTPRINT? ?)
S4       1267   S1(5N)(EXTRACT??? OR PARS??? OR SELECT??? OR PICK??? OR ID-
             ENTIF???? OR IDENTIFICATION)(5N)(COMPONENT? ? OR MODULE? ? OR
             FEATURE? ? OR FUNCTION?)
S5    3044709   (APPLICATION? ? OR PROGRAM? ? OR CODE OR SOFTWARE OR FILE?
             ? OR OS OR OPERATING()SYSTEM)(3N)(DEVELOP? OR GENERAT? OR CRE-
             AT??? OR DESIGN?? OR DESIGNING OR BUILD? OR CONSTRUCT? OR PRO-
             DUCE? ? OR PRODUCTION OR AUTHOR??? OR WRIT???)
S6       4853   S2:S4(50N)S5
S7        178   S2(50N)S5
S8        262   S4(50N)S5
S9     750444   S5(3N)(KIT? ? OR TOOLKIT? ? OR PLATFORM? ? OR ENVIRONMENT?
             ? OR TOOL? ? OR PACKAGE? ? OR SYSTEM? ? OR ARCHITECTURE? ? OR
             SOLUTION? ? OR METHOD? ? OR METHODOLOG??? OR MODULE? ? OR FRA-
             MEWORK? ?)
S10      2603   S2:S4(50N)S9
S11        87   S2(50N)S9
S12       122   S4(50N)S9
S13     88080   S1(5N)(COMPONENT? ? OR MODULE? ? OR FEATURE? ? OR FUNCTION?
              ? OR FILE? ? OR LIBRARIES OR LIBRARY OR DLL OR DLLS OR OBJEC-
             T? ? OR API OR APIS)
S14       511   S2:S4(30N)S9(30N)S13
S15       173   (S2 OR S4)(50N)S9(50N)S13
S16       127   (S2 OR S4)(30N)S9(30N)S13
S17        80   RD (unique items)
S18        73   S17 NOT PD>20011201
S19   2840545   (APPLICATION? ? OR PROGRAM? ? OR SOFTWARE OR OS OR OPERATI-
             NG()SYSTEM)(3N)(DEVELOP? OR GENERAT? OR CREAT??? OR DESIGN?? -
             OR DESIGNING OR BUILD? OR CONSTRUCT? OR PRODUCE? ? OR PRODUCT-
             ION OR AUTHOR???)
S20    712829   S19(3N)(KIT? ? OR TOOLKIT? ? OR PLATFORM? ? OR ENVIRONMENT?
              ? OR TOOL? ? OR PACKAGE? ? OR SYSTEM? ? OR ARCHITECTURE? ? OR
              SOLUTION? ? OR METHOD? ? OR METHODOLOG??? OR MODULE? ? OR FR-
             AMEWORK? ?)
S21       115   (S2 OR S4)(30N)S20(30N)S13
S22        66   S18 AND S21
S23    128452   (TRACE? ? OR TRACING OR RESOLV? OR RESOLUTION? ? OR CONNEC-
         T? OR LINK???)(5N)(REFER??? OR REFERENC? OR DEPENDEN? OR ASSO-
         CIAT? OR CORRELAT? OR RELATED)
S24      294   S1(30N)S23(30N)S20
S25        6   S2:S4(30N)S23(30N)S20
S26        4   RD (unique items)
```

22/9/23 (Item 2 from file: 621)
DIALOG(R)File 621:Gale Group New Prod.Annou.(R)

02761111 Supplier Number: 68147151 (THIS IS THE FULLTEXT)
Microsoft Announces Release of 'Whistler Embedded' Beta 1.
PR Newswire, pNA
Dec 18, 2000
Language: English Record Type: Fulltext
Document Type: Newswire; Trade
Word Count: 804
TEXT:

Targeted Beta Version Delivers the First Step Toward Enabling Microsoft NET Client Devices

REDMOND, Wash., Dec. 18 /PRNewswire/ --

Microsoft Corp. (Nasdaq: MSFT) announced today the beta 1 release of the next embedded version of the Microsoft(R) Windows(R) 2000 operating system, currently code-named "Whistler Embedded." "Whistler Embedded" delivers all the richness, innovation and reliability of the next generation of Windows in a componentized version, enabling embedded developers to select specific technology components required for a wide range of unique Windows Powered devices. Beta 1 is focused on delivering embedded client scenarios, including Windows-based terminals, advanced set-top boxes and retail point-of-sale kiosks. With the release of "Whistler Embedded" beta 1 coming only 48 days after release of the "Whistler" beta, Microsoft underscores its commitment to bringing the latest, most innovative Windows technology to the embedded space in a timely manner.

(Photo: http://www.newscom.com/cgi-bin/prnh/20000822/MSFTLOGO)

"Building on the success of Windows NT(R) Embedded 4.0, we are committed to helping embedded developers utilize the 'Whistler' advancements through componentization and a rich embedded tool set," said Bill Veghte, vice president of the Embedded and Appliance Platforms Group at Microsoft. "The release of 'Whistler Embedded' beta 1 is an important milestone because it facilitates feedback from our close customers and industry partners, which is critical to ensuring a reliable and well-tested final release for the fast-growing embedded industry."

"Whistler Embedded" is being developed around three primary design goals: The first is to deliver extensive componentization of the "Whistler" operating system so appliance and device manufacturers have more flexibility and choice from a broad set of features and functionality. The second is to bring the most innovative and cutting-edge Windows technology to the embedded space within 90 days of the release of "Whistler." The third is to deliver a powerful new set of development and authoring tools to help facilitate quicker development and reduce the time to market.

The new development and authoring tools include the following:

-- Target Designer enables embedded developers to select the necessary feature components from a database of all available Windows components, add them to the run-time configuration and **generate** the customized **operating system** image.

-- Component **Designer** enables embedded developers to **design** their own custom **operating system** components and add them to the run-time configuration, and gives developers the capacity to add additional custom applications to the run-time configuration for the flexibility they need to build embedded systems.

-- Database Manager allows embedded developers to import their custom-built operating system components into the "Whistler Embedded" database repository. It also provides additional utilities for database management tasks such as changing server location, viewing database objects, and managing platforms and repositories.

-- Target Analyzer aids developers in the design process by providing system information such as which are the critical device drivers of the intended target system. Target Analyzer will quickly identify dependencies between critical operating system components and the required device drivers for the custom **operating system** image, helping **developers** quickly **build** an **operating system** image that will successfully boot on the target device.

"VenturCom has a strong relationship with Microsoft stemming from our experience with Windows NT Embedded 4.0 as well as with developing products and services, such as RTX 5.0, that extend the platform to provide real-time capabilities," said Michael Dexter-Smith, president and CEO of VenturCom. "The 'Whistler Embedded' beta program will give us the opportunity to develop our expertise on the latest Windows technology, and work closely with Microsoft to provide feedback that results in the best-quality product for our mutual customers."

About the Microsoft Windows Embedded Family and Microsoft

The Microsoft Windows Embedded operating systems and tools provide comprehensive software platforms for building the next generation of intelligent, 32-bit connected Windows Powered devices that demand rich applications and Internet services for a wide range of flexible solutions. In addition, Microsoft offers a wide range of programs and services designed to meet the specific needs of Windows Embedded customers, industry partners and developers. Windows Embedded includes Windows CE 3.0 and Windows NT Embedded 4.0.

Founded in 1975, Microsoft is the worldwide leader in software, services and Internet technologies for personal and business computing. The company offers a wide range of products and services designed to empower people through great software -- any time, any place and on any device.

NOTE: Microsoft, Windows and Windows NT are either registered trademarks or trademarks of Microsoft Corp. in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

PUBLISHER NAME: PR Newswire Association, Inc.
COMPANY NAMES: *Microsoft Corp.
GEOGRAPHIC NAMES: *1USA (United States)
PRODUCT NAMES: *7372000 (Computer Software)
INDUSTRY NAMES: BUS (Business, General); BUSN (Any type of business)
SIC CODES: 7372 (Prepackaged software)
NAICS CODES: 51121 (Software Publishers)
TICKER SYMBOLS: MSFT

22/9/41 (Item 2 from file: 16)
DIALOG(R)File 16:Gale Group PROMT(R)
(c) 2004 The Gale Group. All rts. reserv.

06569722 Supplier Number: 55474332 (THIS IS THE FULLTEXT)
Microsoft goes after OEM market with embedded code. (Windows NT Embedded) (Product Announcement)
Fontana, John
Network World, pNA
August 16, 1999
ISSN: 0887-7661
Language: English Record Type: Fulltext
Article Type: Product Announcement
Document Type: Magazine/Journal; Trade
Word Count: 460
TEXT:

Microsoft officially moved into the embedded operating system market with yesterday's release of Windows NT Embedded 4.0.

The software will be licensed to OEMs for inclusion in such devices as printers, scanners, PBX, ATM switches and integrated voice response systems. The software will not be sold to enterprise network customers.

However, the upshot for enterprise net customers is that devices with an Embedded NT operating system will be easier to integrate into a Windows infrastructure, according to Microsoft officials. Typically, device vendors use general-purpose commodity operating systems in their products.

The software is based on Windows NT 4.0 with Service Pack 5 and includes the entire set of services **APIs** from that operating system. The embedded operating system is different in that it is only for fixed-function devices. Also, OEMs can pick and choose which components of the operating system, such as the kernel or protocol stack, they wish to include in their devices, in essence **creating** a customized **operating system** . Embedded NT also includes features specific to that operating system, including "headless" support for where there is no mouse or keyboard, and diskless support so applications can run from flash memory or a CD-ROM.

Embedded NT also ships with Target Designer, which is used to select the **components** for **building** a customized **operating system** , and **Component Designer** , which helps **developers** integrate **applications** into the operating **system** .

But some are expressing concerns over the embedded operating system given complaints over NT's reliability.

"With a network appliance, I don't want to have to reboot. I want to set it down and have it keep on working," says independent analyst Dave Kearns. Others weaknesses were pointed out, like hot fixes and service packs which come from Microsoft but must be distributed through OEMs.

Potential devices, or appliances, that may use Embedded NT include handheld devices, network hardware, office automation equipment, medical devices, retail sales systems and server appliances such as directory, caching and network-attached storage, according to a report by The Gartner Group.

Microsoft is already busy working on the next version of the software, according to Vince Mendillo, lead product manager for Windows NT Embedded. The Windows 2000 Embedded version will ship 8 to 9 months after Windows 2000 launches, he says. Windows 2000 is expected to ship by the year-end. "Active Directory will become an interesting and valuable tool for managing elements of the embedded operating system," Mendillo says.

Microsoft is positioning the operating system at the middle to high end of the embedded market that is dominated by companies such as Wind River and Integrated Systems.

A host of companies have already aligned with Microsoft, including Hewlett-Packard, Xerox, Lucent and National Semiconductor.

Find out more about Windows NT Embedded 4.0
http://www.microsoft.com/embedded

~End:

PUBLISHER NAME: Network World, Inc.
COMPANY NAMES: *Microsoft Corp.
EVENT NAMES: *336 (Product introduction)
GEOGRAPHIC NAMES: *1USA (United States)
PRODUCT NAMES: *7372502 (Operating Systems)
INDUSTRY NAMES: BUSN (Any type of business); CMPT (Computers and Office Automation)
NAICS CODES: 51121 (Software Publishers)
TRADE NAMES: Microsoft Windows NT Embedded 4.0 (Operating system)
SPECIAL FEATURES: COMPANY

22/9/42 (Item 3 from file: 16)
DIALOG(R)File 16:Gale Group PROMT(R)

04579327 Supplier Number: 46731998 (THIS IS THE FULLTEXT)
Chorus OS focuses on components
Electronic Engineering Times, p20
Sept 23, 1996
ISSN: 0192-1541
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 80
TEXT:

San Jose, Calif. - An architecture that makes it easier to build programs by providing what's called a componentized operating system is the latest advance from Chorus Systems (St. Quentin, France). The new offering, ClassiXr3, lets designers **pick** and choose **components** to **build** an **operating system** that provides just the **features** needed for a given system. In another move, Chorus signed a pact with Green Hills Software Inc. (Santa Barbara, Calif.) that gives Chorus users access to Green Hills' development environment.

```
File   8:Ei Compendex(R) 1970-2004/Sep W4
         (c) 2004 Elsevier Eng.  Info. Inc.
File  35:Dissertation Abs Online 1861-2004/Sep
         (c) 2004 ProQuest Info&Learning
File 202:Info. Sci. & Tech. Abs. 1966-2004/Sep 09
         (c) 2004 EBSCO Publishing
File  65:Inside Conferences 1993-2004/Oct W1
         (c) 2004 BLDSC all rts. reserv.
File   2:INSPEC 1969-2004/Sep W4
         (c) 2004 Institution of Electrical Engineers
File  94:JICST-EPlus 1985-2004/Sep W1
         (c)2004 Japan Science and Tech Corp(JST)
File 483:Newspaper Abs Daily 1986-2004/Oct 05
         (c) 2004 ProQuest Info&Learning
File   6:NTIS 1964-2004/Sep W4
         (c) 2004 NTIS, Intl Cpyrght All Rights Res
File 144:Pascal 1973-2004/Sep W4
         (c) 2004 INIST/CNRS
File 434:SciSearch(R) Cited Ref Sci 1974-1989/Dec
         (c) 1998 Inst for Sci Info
File  34:SciSearch(R) Cited Ref Sci 1990-2004/Sep W4
         (c) 2004 Inst for Sci Info
File  99:Wilson Appl. Sci & Tech Abs 1983-2004/Aug
         (c) 2004 The HW Wilson Co.
File 583:Gale Group Globalbase(TM) 1986-2002/Dec 13
         (c) 2002 The Gale Group
File 266:FEDRIP 2004/Jul
         Comp & dist by NTIS, Intl Copyright All Rights Res
File  95:TEME-Technology & Management 1989-2004/Jun W1
         (c) 2004 FIZ TECHNIK
File 438:Library Lit. & Info. Science 1984-2004/Aug
         (c) 2004 The HW Wilson Co

Set     Items   Description
S1     268585   OS OR OPERATING()(SYSTEM? ? OR ENVIRONMENT? ?)
S2        172   S1(5N)(REDUC???? OR SHRINK??? OR LOWER??? OR DECREAS??? OR
             TRIM???? OR CONTRACT? OR CONDENS?)(5N)(SIZE OR SCALE OR FEATU-
             RE? ? OR FUNCTION? OR MAGNITUDE)
S3       2867   S1(7N)(SUBSET? ? OR SIMPLIF? OR SMALL??? OR COMPACT? OR MO-
             DULAR? OR FOOTPRINT? ?)
S4        202   S1(5N)(EXTRACT??? OR PARS??? OR SELECT??? OR PICK??? OR ID-
             ENTIF???? OR IDENTIFICATION)(5N)(COMPONENT? ? OR MODULE? ? OR
             FEATURE? ? OR FUNCTION?)
S5    1146893    (APPLICATION? ? OR PROGRAM? ? OR CODE OR SOFTWARE OR FILE?
              ? OR OS OR OPERATING()SYSTEM)(3N)(DEVELOP? OR GENERAT? OR CRE-
             AT? OR DESIGN?? OR DESIGNING OR BUILD? OR CONSTRUCT? OR PRODU-
             C? OR AUTHOR??? OR WRIT???)
S6        918   S2:S4 AND S5
S7         36   S2 AND S5
S8        841   S3 AND S5
S9         52   S4 AND S5
S10        88   S7 OR S9
S11        76   RD (unique items)
S12        68   S11 NOT PY=2002:2004
S13     10542   S1(5N)(COMPONENT? ? OR MODULE? ? OR FEATURE? ? OR FUNCTION?
              ? OR FILE? ? OR LIBRARIES OR LIBRARY OR DLL OR DLLS OR OBJEC-
             T? ?)
S14       159   S8 AND S13
S15    228242   S5(3N)(KIT? ? OR TOOLKIT? ? OR PLATFORM? ? OR ENVIRONMENT?
             ? OR TOOL? ? OR PACKAGE? ? OR SYSTEM? ? OR ARCHITECTURE? ? OR
             SOLUTION? ? OR METHOD? ? OR METHODOLOG??? OR MODULE? ? OR FRA-
             MEWORK? ?)
S16       115   S14 AND S15
S17        84   RD (unique items)
S18        72   S17 NOT (S12 OR PY=2002:2004)
S19    114298   (TRACE? ? OR TRACING OR RESOLV? OR RESOLUTION? ? OR CONNEC-
             T? OR LINK???)(5N)(REFER??? OR REFERENC? OR DEPENDEN? OR ASSO-
             CIAT? OR CORRELAT? OR RELATED)
```

```
S20          9   S2:S4 AND S19
S21          7   RD (unique items)
S22        541   S15 AND S19
S23        762   SOFTWARE(2N)DEVELOP?(2N)(KIT OR KITS OR TOOLKIT? ?)
S24          2   S23 AND S19
S25        517   AU=(BOGDAN S? OR BENTLEY K? OR BOGDAN, S? OR BENTLEY, K?)
S26          6   S25 AND S5
S27          5   RD (unique items)
```

12/5/9 (Item 9 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

03347745 E.I. Monthly No: EI9112148276
Title: LOCUS operating system, a transparent system.
Author: Thiel, Greg
Corporate Source: Locus Computing Corp, Inglewood, CA, USA
Source: Computer Communications v 14 n 6 Jul-Aug 1991 p 336-346
Publication Year: 1991
CODEN: COCOD7 ISSN: 0140-3664
Language: English
Document Type: JA; (Journal Article) Treatment: A; (Applications)
Journal Announcement: 9112

Abstract: The primary system components and architecture of the LOCUS operating system and the Transparent Computing Facility are identified. The current level of TCF functionality is described. References to the LOCUS OS are used in areas where the fundamental concepts are described. The initial two sections outline the vision of the LOCUS **Operating System** , and provide definitions of key terms. The third section **identifies** the **functionality** provided by IBM's Transparent Computing Facility. The fourth section describes several of the key algorithms and basic system structure, and the last two sections provide insight into future directions for the technology, and summarize the material presented. (Author abstract) 10 Refs.

Descriptors: COMPUTER OPERATING SYSTEMS--* **Design** ; COMPUTER **SOFTWARE** --Software Engineering
Identifiers: LOCUS OPERATING SYSTEMS; TRANSPARENT COMPUTING
Classification Codes:
722 (Computer Hardware); 723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

12/5/11 (Item 11 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

01914149 E.I. Monthly No: EIM8512-080663
Title: PROMISING APPLICATION OF MICROPROGRAMMING: VERTICAL MIGRATION.
Author: Ambrozy, G.
Corporate Source: Central Research Inst for Physics, Budapest, Hung
Conference Title: Proceedings of the Third Symposium on Microcomputer and Microprocessor Application.
Conference Location: Budapest, Hung Conference Date: 19831018
Sponsor: Scientific Soc for Telecommunication, Budapest, Hung; John V. Neumann Soc for Computing Science, Budapest, Hung; Scientific Soc of Measurement & Automation, Budapest, Hung
E.I. Conference No.: 05766
Source: v 1. Publ by OMIKK Technoinform, Budapest, Hung p 87-100
Publication Year: 1983
ISBN: 963-592-295-7
Language: English
Document Type: PA; (Conference Paper)
Journal Announcement: 8512

Abstract: One of the significant applications of microprogramming is vertical migration - a technique by which computer performance can be increased, computer architectures can be redefined and restructured. In a project in the Central Research Institute for Physics some tools helping vertical migration **application** were **developed** . One of the objectives was the thorough analysis of operating system behavior by means of a firmware monitor. Based on the results of monitoring the **operating system** kernel several **functions** were **selected** as candidates for vertical migration. After the practical implementation of migration the predicted performance improvements were checked. 8 refs.

Descriptors: *COMPUTER ARCHITECTURE--*Microprogramming; COMPUTER OPERATING SYSTEMS--Analysis; COMPUTERS--Performance
Identifiers: OPERATING SYSTEM BEHAVIOR; VERTICAL MIGRATION APPLICATION
Classification Codes:

722 (Computer Hardware); 723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

12/5/12 (Item 12 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

01723072 E.I. Monthly No: EI8501001428 E.I. Yearly No: EI85021103
Title: DEDICATED, DISTRIBUTED AND PORTABLE OPERATING SYSTEMS: A STRUCTURING CONCEPT.
Author: Valdorf, G.
Corporate Source: Siemens AG, Munich, West Ger
Source: Software - Practice and Experience v 14 n 11 Nov 1984 p 1079-1093
Publication Year: 1984
CODEN: SPEXBL ISSN: 0038-0644
Language: ENGLISH
Document Type: JA; (Journal Article) Treatment: X; (Experimental)
Journal Announcement: 8501
Abstract: A proposal is presented for a structural concept permitting the economic and controlled production of voluminous and complex software systems. This concept is explained taking the **design** of **selected operating system functions** as examples. The attributes 'economic' and 'controlled' are explained. 25 refs.
Descriptors: *COMPUTER OPERATING SYSTEMS--*Distributed
Identifiers: OPERATING SYSTEM STRUCTURE; VIRTUAL MACHINE; MONITOR INCREMENTAL MACHINE DESIGN
Classification Codes:
722 (Computer Hardware); 723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

12/5/14 (Item 1 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2004 ProQuest Info&Learning. All rts. reserv.

01818896 ORDER NO: AADAA-I3003667
Piglet: An operating system for network appliances
Author: Muir, Stephen James
Degree: Ph.D.
Year: 2001
Corporate Source/Institution: University of Pennsylvania (0175)
Supervisor: Jonathan M. Smith
Source: VOLUME 62/02-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 936. 106 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984
ISBN: 0-493-13111-6

Advances in the performance of commodity hardware and acceptance of open-source software have recently led to the increased use of systems based upon a combination thereof as network appliances. The principal thesis of this dissertation is that such appliances can operate more efficiently if their **operating system** is **designed** specifically for that task. I present <italic>Piglet</italic> as a novel **design** for such an **operating system** , and describe its implementation and evaluation in such a context.

The core of the Piglet architecture is the <italic>Active Kernel</italic>. A dedicated kernel processor provides concurrency between kernel and applications and enables asynchronous shared-memory communication. The use of shared objects for all communication between applications and the kernel permits Piglet to offer much more efficient mechanisms for the invocation of system services than a conventional kernel.

The experimental results presented herein demonstrate how these fundamental kernel **features** make Piglet a more efficient **operating system** for network appliances: **reduced** latency for individual operations and increased aggregate data throughput. In addition to detailing microbenchmark results to support those claims, I show how an existing web server application can readily make use of those enhancements to achieve a performance boost over standard operating systems.

Finally, this dissertation examines, both quantitatively and qualitatively, some of the challenges and problems presented by the implementation of the Piglet architecture, and proposes solutions and/or architectural changes to address those concerns.

12/5/15 (Item 2 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

01779817 ORDER NO: AADAA-I9992529
Real-time software for control and robotic applications
Author: Costescu, Nicolae Peter
Degree: Ph.D.
Year: 2000
Corporate Source/Institution: Clemson University (0050)
Adviser: Darren M. Dawson
Source: VOLUME 61/10-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 5461. 81 PAGES
Descriptors: ENGINEERING, ELECTRONICS AND ELECTRICAL ; COMPUTER SCIENCE
Descriptor Codes: 0544; 0984
ISBN: 0-493-00409-2

This doctoral dissertation describes real-time software systems for use in control and robotic applications. The body of this dissertation focuses on three systems developed at Clemson University: (i) QMotor 2.0, a single-processor based system for implementing and tuning control programs in <italic>C</italic>, (ii) Motor 3.0, an object-oriented system for implementing and tuning control programs in <italic>C</italic>++, and (iii) QRobot, a single-processor based control system for Puma robotic manipulators.

Traditionally, digital control programs were implemented using a heterogeneous multi-processor architecture, typically consisting of a <italic>host</italic> general purpose computer (PC, Mac, SGI, <italic>etc</italic>.) and an embedded digital signal processor (DSP) based single board computer (SBC). This architecture was required because general purpose computers (GPCs) either did not have enough processing power to compute a control law at high frequencies, and/or did not run real-time operating systems that would allow them to execute the control programs deterministically. The first chapter describes QMotor 2.0, a system that integrates modern PC CPUs and a real-time **operating system** to **produce** a uniprocessor architecture with greater **functionality** and **lower** cost and complexity than the traditional multi-processor <italic>host</italic>/<italic> DSP</italic> architecture.

The second chapter describes QMotor 3.0, the successor to QMotor 2.0. As QMotor 2.0 was used constantly at Clemson University, many shortcomings became apparent. The field of Control Engineering requires great flexibility in implementing experiments. New hardware interfaces must constantly be added to take advantage of the latest sensors and actuators. The functional programming design of QMotor 2.0 limited the ability to re-use code, and to easily modify and maintain the code. QMotor 3.0 is a complete redesign of QMotor 2.0, using object-oriented programming (OOP) principles throughout, and implemented in the <italic>C</italic>++ programming language.

The third chapter describes QRobot, a robot control system that builds on the concepts and technology of QMotor 3.0. Traditionally robot control systems are composed of closed, multiprocessor architectures. The closed architecture ensures that users do not have access to the control algorithms used to drive the robot. This prevents users from implementing more sophisticated control algorithms that become possible as CPUs become more powerful. It also prevents the integration of new sensors (such as force/torque sensors and video cameras) into the control loop. QRobot demonstrates how a closed, proprietary, firmware-based robot control system can be replaced with an open, PC-based control system, lowering cost and complexity, and increasing flexibility and power.

12/5/16 (Item 3 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2004 ProQuest Info&Learning. All rts. reserv.

01681018 ORDER NO: AADNQ-34638
A DYNAMICALLY RECONFIGURABLE AND EXTENSIBLE OPERATING SYSTEM
Author: VEITCH, ALISTAIR CRAIG
Degree: PH.D.
Year: 1998
Corporate Source/Institution: THE UNIVERSITY OF BRITISH COLUMBIA (CANADA) (2500)
Adviser: NORMAN HUTCHINSON
Source: VOLUME 59/12-B OF DISSERTATION ABSTRACTS INTERNATIONAL. PAGE 6387. 144 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984
ISBN: 0-612-34638-2

Operating systems are constantly getting more complex in the functionality they support, due to the increasing demands made by modem hardware and software innovations. Basing the kernel design on co-operating and modular services incorportating a flexible communications infrastructure with run-time binding makes the operating system dynamically configurable and extensible. These features aid in the management of system complexity, while also resulting in several software engineering and performance benefits.

Configurability gives the **operating system designer** and implementor the freedom to build a large number of components, which can be composed into different configurations depending upon the final system requirements. System components can be built and debugged in a user address space, and then transparently migrated into the kernel address space for performance once they have been demonstrated correct. This removes one of the major obstacles to developing kernel services, that of the necessity to reboot the system after each change to the service code. The system administrator can also reconfigure the system, providing similar advantages, and allowing dynamic system upgrades to be made, **reducing** system downtime.

Extensibility lets new **functionality** be integrated into the **operating system** . This can be done on an application specific basis. This enables the **development** of in-kernel **applications** in cases where high performance is required, such as for dedicated file servers. It is also possible for applications to interpose specialised kernel services, allowing them to dramatically increase their performance and aggregate system throughput when the default system policies are ill-matched to their behaviour.

The Kea **operating system** has been **designed** and implemented to be dynamically configurable and extensible. The design of the system features that make these features possible are described. Experimental results are shown that demonstrate that Kea offers comparable performance to a traditional operating system on the same hardware, and that extensibility can be used to increase performance for selected applications.

12/5/21 (Item 1 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

7314798 INSPEC Abstract Number: C2002-08-6150J-014
Title: Size reduction **of multi-purpose** operating systems **by restricting APIs**
Author(s): Moriwaka, K.; Kitasuka, T.; Nakanishi, T.; Fukuda, A.
Author Affiliation: Graduate Sch. of Inf. Sci. & Electr. Eng., Kyushu Univ., Fukuoka, Japan
Conference Title: Proceedings of the IASTED International Conference Applied Informatics International Symposium on Software Engineering, Databases, and Applications p.103-7

Editor(s): Hamza, M.H.
Publisher: ACTA Press, Anaheim, CA, USA
Publication Date: 2001 Country of Publication: USA   iv+526 pp.
ISBN: 0 88986 322 9   Material Identity Number: XX-2002-00971
Conference Title: Proceedings of the IASTED International Conference Applied Informatics. International Symposium on Software Engineering, Databases, and Applications
Conference Sponsor: IASTED
Conference Date: 18-21 Feb. 2002   Conference Location: Innsbruck, Austria
Language: English   Document Type: Conference Paper (PA)
Treatment: Practical (P)

Abstract: We propose a technique that reduces the memory footprint of multi-purpose operating systems (OSs) for application specificity. Application programs use a subset of APIs (application program interfaces) which implies which functions of the **OS** are used. We **generate** a function call graph of the OS and library, which can associate APIs with OS functions. In this paper, methods to analyze **application programs** and to **generate** function call graphs of OSs are described. We use MINIX and Linux as sample target OSs, and the init program as the target application. Experimental results show that MINIX OS has about 20% unused code for the init program. (10 Refs)

Subfile: C
Descriptors: application program interfaces; operating systems (computers)
Identifiers: multi-purpose operating systems; size reduction; memory footprint; API restriction; function call graphs; MINIX; Linux; init program
Class Codes: C6150J (Operating systems); C6150E (General utility programs)
Copyright 2002, IEE

12/5/23   (Item 3 from file: 2)
DIALOG(R)File   2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6618198   INSPEC Abstract Number: C2000-07-6150J-016
Title: Microsoft's Windows CE: why it can compete
Author(s): Yourdon, E.
Journal: Cutter IT Journal   vol.13, no.6   p.27-31
Publisher: Cutter Inf. Corp,
Publication Date: June 2000 Country of Publication: USA
ISSN: 1048-5600
SICI: 1048-5600(200006)13:6L.27:MWC;1-P
Material Identity Number: G495-2000-007
Language: English   Document Type: Journal Paper (JP)
Treatment: Practical (P)

Abstract: In the new world of the embedded system and wireless information device, Microsoft is offering a scaled-down version of its Windows technology known as Windows CE. Actually, scaled down is a misnomer, since one of the objectives of CE is to provide an environment for hosting **applications** that were originally **developed** for desktop PCs; to accomplish this, Microsoft has provided a modular **operating system** **created** specifically for embedded systems, allowing developers to **select** only the **functionality** they need for specific embedded designs. Thus, for example, there are versions of such familiar applications as Word, Excel, and PowerPoint that run on commercially popular PDAs. On the other hand, scaled down is appropriate, since the PDA user is provided with a subset of the full capabilities of Word, a subset of Excel, and a subset of PowerPoint. (0 Refs)

Subfile: C
Descriptors: business data processing; embedded systems; notebook computers; operating systems (computers)
Identifiers: Microsoft Windows CE; embedded system; wireless information device; desktop PC; operating system; Word; Excel; PowerPoint; personal digital assistant
Class Codes: C6150J (Operating systems); C7100 (Business and

administration); C5430 (Microcomputers)
Copyright 2000, IEE

12/5/24 (Item 4 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6473797 INSPEC Abstract Number: C2000-02-6150J-039
Title: EMERALDS: a small-memory real-time microkernel
Author(s): Zuberi, K.M.; Pillai, P.; Shin, K.G.
Author Affiliation: Michigan Univ., MI, USA
Journal: Operating Systems Review Conference Title: Oper. Syst. Rev. (USA) vol.33, no.5 p.277-91
Publisher: ACM,
Publication Date: Dec. 1999 Country of Publication: USA
CODEN: OSRED8 ISSN: 0163-5980
SICI: 0163-5980(199912)33:5L.277:ESMR;1-8
Material Identity Number: O043-2000-001
Conference Title: 17th ACM Symposium on Operating Systems Principles
Conference Date: 12-15 Dec. 1999 Conference Location: Kiawah Island Resort, SC, USA
Language: English Document Type: Conference Paper (PA); Journal Paper (JP)
Treatment: Practical (P)

Abstract: EMERALDS (Extensible Microkernel for Embedded ReALtime, Distributed Systems) is a real time microkernel designed for small-memory embedded applications. These applications must run on slow (15-25 MHz) processors with just 32-128 kbytes of memory, either to keep production costs down in mass-produced systems or to keep weight and power consumption low. To be feasible for such applications, the OS must not only be small in size (less than 20 kbytes), but also have low-overhead kernel services. Unlike commercial embedded OSs which rely on carefully-crafted code to achieve efficiency, EMERALDS takes the approach of re- **designing** the basic **OS** services of task scheduling, synchronization, communication, and system call mechanism by using characteristics found in small-memory embedded systems, such as small code size and a priori knowledge of task execution and communication patterns. With these new schemes, the overheads of various **OS** services are **reduced** 20-40% without compromising any **OS functionality** . (41 Refs)

Subfile: C
Descriptors: distributed processing; embedded systems; operating system kernels; scheduling; storage management
Identifiers: EMERALDS; small-memory real time microkernel; Extensible Microkernel for Embedded ReALtime Distributed Systems; real time microkernel; small-memory embedded applications; production costs; mass-produced systems; power consumption; low-overhead kernel services; commercial embedded OSs; basic OS services; task scheduling; system call mechanism; small-memory embedded systems; small code size; a priori knowledge; task execution; communication patterns; OS functionality
Class Codes: C6150J (Operating systems); C6150N (Distributed systems software); C6120 (File organisation)
Copyright 2000, IEE

12/5/25 (Item 5 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6384659 INSPEC Abstract Number: B1999-12-8520B-004, C1999-12-7445-008
Title: Designing an OSEK development **environment for automotive** applications
Author(s): Foster, N.
Journal: Real-Time Magazine p.33-6
Publisher: Real-Time Consult,
Publication Date: July-Sept. 1999 Country of Publication: Belgium
CODEN: RTMAFD ISSN: 1018-0303
Material Identity Number: E387-1999-004

Language: English   Document Type: Journal Paper (JP)
Treatment: Practical (P)

Abstract: Today's automotive manufacturers are challenged with replacing traditional electromechanical systems with complex embedded systems. At present, a variety of software environments are used by suppliers, both off the shelf and bespoke, making it very difficult to change CPU suppliers or reuse the **software** in a new **generation** of products. The OSEK/VDX standard is emerging as a possible answer, with standardised interfaces and **functionality** , the **OS** supports re-usability and portability of application software modules and therefore **reduces** development cost. Developing complex OSEK-based systems requires additional **OS** aware tools to minimize important **development** time. **OS** awareness is not a trivial issue and requires tight integration between the development environment of OSEK, the debugging tools, and means of communicating with such tools. One solution is a totally Integrated Development Environment (IDE) incorporating tools and OS from the same vendor delivering OSEK awareness and connections to high level case tools. (5 Refs)

Subfile: B C

Descriptors: automotive electronics; development systems; embedded systems; programming environments; traffic engineering computing

Identifiers: OSEK development environment; automotive applications; embedded systems; software environments; Integrated Development Environment

Class Codes: B8520B (Automobile electronics); C7445 (Traffic engineering computing); C5250 (Microcomputer techniques); C6115 (Programming support)

Copyright 1999, IEE

12/5/26   (Item 6 from file: 2)

DIALOG(R)File   2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6129960   INSPEC Abstract Number: C1999-02-6150J-012

Title: The Windows CE build process

Author(s): Havewala, A.

Journal: Dr. Dobb's Journal   vol.23, no.8   p.50, 52, 54, 56-7, 59, 91-2

Publisher: Miller Freeman,
Publication Date: Aug. 1998 Country of Publication: USA
CODEN: DDJSDM ISSN: 1044-789X
SICI: 1044-789X(199808)23:8L.50:WBP;1-B
Material Identity Number: B719-1998-012
Language: English   Document Type: Journal Paper (JP)
Treatment: Practical (P)

Abstract: Windows CE is an operating system that can be separated into any number of components, which may or may not be bundled with each other. Consequently. when you create a unique hardware platform and use Windows CE as the **operating system** , you can **pick** and choose from the available **components** what to include in your custom version of Windows CE. The Windows CE build process specifies which components should go into a custom version, which is then built in the form of a binary **file** . The components and **build** process are available from Microsoft in a development kit called the Windows CE Embedded Toolkit (ETK) that runs on Windows NT. The ETK currently comes with three CD-ROMs. One contains the Windows CE **software development** kit (SDK) and the Visual C++ toolkit for CE. The other two have the ETK, which allows you to integrate your drivers with the kernel, write a hardware adaptation layer (HAL) for your hardware, and create a custom version of Windows CE for your board. Because of the need to create a flexible build process, Microsoft has created an extensive and complex build process that lets you specify how a custom version of Windows CE must be built for a specific hardware platform. Typically, after new hardware is designed and available, you choose one of the demo projects in the Windows CE ETK and modify it to suit your needs. This approach of treating the build process as a black box is fraught with the potential for mistakes, which result in unpredictable problems. Such problems can, quite simply, be the result of bad builds. In this article, I focus on the CEPC platform of the Windows CE Alder preview ETK. (0 Refs)

Subfile: C

Descriptors: development systems; graphical user interfaces; operating

systems (computers); software tools
Identifiers: Microsoft Windows CE; build process; operating system; custom version; binary file; Windows CE Embedded Toolkit; Windows NT; CD-ROMs; **software development** kit; Visual C++ toolkit; drivers; kernel; hardware adaptation layer; CEPC platform; Windows CE Alder preview ETK
Class Codes: C6150J (Operating systems); C6180G (Graphical user interfaces); C6115 (Programming support)
Copyright 1999, IEE

12/5/29 (Item 9 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

4512238 INSPEC Abstract Number: C9312-6150N-023
Title: A message-passing picokernel
Author(s): Jones, S.
Journal: Embedded Systems Programming vol.6, no.3 p.14-17, 19-23
Publication Date: March 1993 Country of Publication: USA
CODEN: EYPRE4 ISSN: 1040-3272
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)
Abstract: Although real-time **operating systems** differ greatly in their **functionality** and architectures, at the **lower** levels they are more similar than different. For those developers who are concerned about portability, the author presents an interface scheme that promises easier migration for real-time applications. He has experimented with a simple microkernel design that explored message-passing kernel architectures. The results are two-fold. First, the author investigates how message-passing kernels work, giving an insight into how one can use the same techniques in embedded **applications** . Second, he **develops** a message-passing kernel that can be used in 80*86-based real-time embedded designs. For times when you need many different coroutines exchanging messages, this simple kernel is the perfect solution. (4 Refs)
Subfile: C
Descriptors: message passing; operating systems (computers); real-time systems; software portability
Identifiers: applications migration; 80*86 based designs; picokernel; real-time operating systems; portability; interface scheme; microkernel design; message-passing kernel architectures; embedded applications; coroutines
Class Codes: C6150N (Distributed systems)

12/5/32 (Item 12 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

04265271 INSPEC Abstract Number: C9212-6150N-012
Title: Modularity and interfaces in micro-kernel design and implementation: a case study of CHORUS on the HP PA-RISC
Author(s): Walpole, J.; Inouye, J.; Konuru, R.
Author Affiliation: Dept. of Comput. Sci. & Eng., Oregon Graduate Inst. of Sci. & Technol., Corvallis, OR, USA
Conference Title: Proceedings of the USENIX Workshop on Micro-Kernels and Other Kernel Architectures p.71-82
Publisher: USENIX Assoc, Berkeley, CA, USA
Publication Date: 1992 Country of Publication: USA 303 pp.
Conference Date: 27-28 April 1992 Conference Location: Seattle, WA, USA
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: The key concept that distinguishes micro-kernel operating systems from their macro-kernel counterparts is modularity. Micro-kernels implement **operating system** functionality in well-defined **modules** with clearly **identified** interfaces between them. Proponents of this modular approach to **operating system design** claim that it offers advantages in the areas of portability, correctness, protection, extensibility, and

reconfigurability for distributed architectures. If micro-kernels are to gain wider acceptance however, it is important to ensure that these benefits of modularity can be attained without incurring significant performance degradation when compared to macro-kernels. The paper explores the relationship between modularity and performance by examining an implementation of the CHORUS micro-kernel operating system on the Hewlett-Packard PA-RISC workstation. It outlines the key interfaces in CHORUS and studies the architectural assumptions implicit in these interfaces. (12 Refs)

Subfile: C

Descriptors: network operating systems

Identifiers: CHORUS; HP PA-RISC; micro-kernel operating systems; modularity; Hewlett-Packard PA-RISC workstation

Class Codes: C6150N (Distributed systems)

12/5/36 (Item 16 from file: 2)

DIALOG(R)File 2:INSPEC

(c) 2004 Institution of Electrical Engineers. All rts. reserv.

03322010 INSPEC Abstract Number: C89021095

Title: At the core: an API comparison

Author(s): Morris, R.R.; Brooks, W.E.

Author Affiliation: Appl. Syst. Technol. Inc., Ft. Lauderdale, FL, USA

Journal: PC Tech Journal vol.6, no.12 p.62-77

Publication Date: Dec. 1988 Country of Publication: USA

CODEN: PCTJEZ ISSN: 0738-0194

Language: English Document Type: Journal Paper (JP)

Treatment: Practical (P)

Abstract: The features and API services provided by Unix and OS/2 are numerous and wide-ranging. API services are those that allow applications to interface with the operating system; the API is actually the package of the many system services that the operating system makes available to a **program** and the techniques **developers** use to call them. Comparing selected services-such as memory management, tasking, file I/O, semaphores, pipes, and queues-is an excellent starting point for discerning similarities and differences in the two **operating systems** . The authors focus on **selected functional** groups of internal services provided by each system: memory management; task control; interprocess communications (IPC), including memory sharing, semaphores, queues, and pipes; file and device-control I/O; and library sharing. Although these selected categories do not represent the universe of operating system services, they are the ones that are most frequently used and constitute a good basis for comparing and contrasting the APIs of Unix and OS/2. (0 Refs)

Subfile: C

Descriptors: operating systems (computers); Unix

Identifiers: API services; Unix; OS/2; operating system; tasking; file I/O; memory management; task control; interprocess communications; memory sharing; semaphores; queues; library sharing

Class Codes: C6150J (Operating systems)

12/5/39 (Item 19 from file: 2)

DIALOG(R)File 2:INSPEC

(c) 2004 Institution of Electrical Engineers. All rts. reserv.

02320142 INSPEC Abstract Number: C84045814

Title: Shrinking VMS

Author(s): Morse, K.D.

Journal: Datamation vol.30, no.11 p.95-100

Publication Date: 15 July 1984 Country of Publication: USA

CODEN: DTMNAT ISSN: 0011-6963

Language: English Document Type: Journal Paper (JP)

Treatment: Practical (P)

Abstract: Describes how Digital Equipment Corp. miniaturized a large machine **operating system** into micro size. Digital's design objective for MicroVMS was to **shrink** the widely used virtual memory system (VMS) **operating system** - **developed** along with the VAX-11/780

superminicomputer-so it could perform with MicroVAX's miniaturized hardware, yet support true VAX architecture. Departing again from usual computer engineering practice, the MicroVMS operating system and MicroVAX CPU hardware were also developed in parallel, with constant coordination between the **software** and hardware **design** teams. (0 Refs)
  Subfile: C
  Descriptors: operating systems (computers)
  Identifiers: DEC; Digital Equipment Corp.; operating system; design; MicroVMS; virtual memory system; VMS; miniaturized hardware; VAX architecture; MicroVAX
  Class Codes: C6150J (Operating systems)

12/5/40 (Item 20 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

01671504 INSPEC Abstract Number: C81014961
Title: A structured operating system
  Author(s): McKeag, R.M.
  Author Affiliation: Dept. of Computer Sci., Queen's Univ. of Belfast, Belfast, UK
  Book Title: On the **construction** of **programs** p.107-51
  Editor(s): McKeag, R.M.; Macnaghten, A.M.
  Publisher: Cambridge, Cambridge, UK
  Publication Date: 1980 Country of Publication: UK 422 pp.
  ISBN: 0 521 23090 X
  Language: English Document Type: Book Chapter (BC)
  Treatment: Practical (P)
  Abstract: These notes apply the principles of structured programming to the **design** of a small **operating system** . The emphasis is on structural aspects rather than on **operating system** techniques. The notes begin by specifying the system, **identifying** the principal **components** of the program and defining the interfaces between those components. Then each component is examined in turn to determine the structure and programming techniques necessary to complete it. (0 Refs)
  Subfile: C
  Descriptors: operating systems (computers); structured programming
  Identifiers: structured programming; design; small operating system; specifying; interfaces; programming techniques
  Class Codes: C6150J (Operating systems)

12/5/44 (Item 1 from file: 94)
DIALOG(R)File 94:JICST-EPlus
(c)2004 Japan Science and Tech Corp(JST). All rts. reserv.

04989299 JICST ACCESSION NUMBER: 01A0821024 FILE SEGMENT: JICST-E
 Size Reduction **of Multi-purpose** Operating Systems **for Application-specific.**
MORIWAKA KAZUO (1); FUKUDA AKIRA (1); NAKANISHI TSUNEO (2)
(1) Kyushu Univ.; (2) Advanced Inst. Sci. and Technol., Nara
Denshi Joho Tsushin Gakkai Gijutsu Kenkyu Hokoku(IEIC Technical Report (Institute of Electronics, Information and Communication Enginners), 2001, VOL.101,NO.151(CPSY2001 14-24), PAGE.1-7, FIG.1, TBL.2, REF.10
JOURNAL NUMBER: S0532BBG
UNIVERSAL DECIMAL CLASSIFICATION: 681.3.066
LANGUAGE: Japanese COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Journal
ARTICLE TYPE: Original paper
MEDIA TYPE: Printed Publication
ABSTRACT: When employing a multi-purpose **OS** (**Operating System**) for an embedded system, its **size reduction** is one of inevitable problems. This paper describes the **size reduction** of a multi-purpose **OS** for application-specific system by deleting the unused code of the OS for the application programs. We study two cases where unit of deletion is statement of program language, or procedure. We try how much **size reduction** can be performed by using an existing **OS** . The results show

10%-30% of it is **reduced** under the condition where only the 'init' **program** is activated. (**author** abst.)
DESCRIPTORS: operating system; size effect; programming language; sentence; function(mathematics); UNIX
IDENTIFIERS: MINIX; init
BROADER DESCRIPTORS: system program; computer program; software; effect; formal language; language; mapping(mathematics)
CLASSIFICATION CODE(S): JD03020J

12/5/46 (Item 3 from file: 94)
DIALOG(R)File 94:JICST-EPlus

04247563 JICST ACCESSION NUMBER: 99A0308994 FILE SEGMENT: JICST-E
Development **of Discrete** Application **Package System.**
KANNO NOBUAKI (1); YOMOGIDA MAKOTO (1); NOJIRI YASUAKI (1); TANAKA YOSHINORI (1); KINOSHITA TAKUYA (1); HONMA SATOSHI (1); IIDA YOUSUKE (1)
(1) Yamatakesangyoushisutemu
Savemation Rev, 1999, VOL.17,NO.1, PAGE.46-52, FIG.6, TBL.2
JOURNAL NUMBER: L0532AAX
UNIVERSAL DECIMAL CLASSIFICATION: 681.58
LANGUAGE: Japanese COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Journal
ARTICLE TYPE: Commentary
MEDIA TYPE: Printed Publication
ABSTRACT: An application package system targeting the special discrete field for downstream processes was **developed** . This Discrete **Application** Package System(DAPS) collects large volumes of data at high speed for controlled equipment and facilities and provides various monitoring and control **functions** required at manufacturing sites. Windows-NT was **selected** as the **OS** for the station, and Accessas the database for the standard data storage. The system is extremely open, as it supports OPC(OLE for Process Control) and I/O data access from the outside is possible. (author abst.)
DESCRIPTORS: instrumentation; data collection system; supervisory control system; LAN; production system; programmable controller; process control; relational data base
BROADER DESCRIPTORS: computer application system; system; monitor; equipment; computer network; communication network; information network ; network; control equipment; control; database
CLASSIFICATION CODE(S): IC02010N

12/5/63 (Item 1 from file: 99)
DIALOG(R)File 99:Wilson Appl. Sci & Tech Abs

2398872 H.W. WILSON RECORD NUMBER: BAST01069227
Internet appliances: software frameworks are the key
Wong, William;
Electronic Design v. 49 no13 (June 18 2001) p. 74-80
DOCUMENT TYPE: Feature Article ISSN: 0013-4872 LANGUAGE: English
RECORD STATUS: Corrected or revised record

ABSTRACT: Successful Internet appliances need application connectivity provided by software, as well as network hardware and TCP/IP stacks. A web server, or browser, loaded on to a device only provides limited interaction between devices such as web pads, cell phones, and PCs. Automated connection is required for Internet devices such as MP3 storage devices, which act as both client and server. **Operating system** and framework **developers** are now delivering location and **identification features** such as Jini and Universal Plug and Play, which automate connections between the Internet appliances.

DESCRIPTORS: Information appliances; Extensible Markup Language (Computer language); Hypertext Markup Language (Computer language);

12/5/67 (Item 1 from file: 95)
DIALOG(R)File 95:TEME-Technology & Management

00746705 E94026091021
Protocol modularity in systems for managing replicated data
(Protokollmodularitaet in Systemen zur Verwaltung wiederholter Daten)
Mishra, S; Peterson, LL; Schlichting, RD
Univ. of Arizona, Tucson, USA; Univ. of California, San Diego, USA
2nd Workshop on the Management of Replicated Data, Monterey, USA, Nov 12-13, 19921992
Document type: Conference paper Language: English
Record type: Abstract
ISBN: 0-8186-3170-8

ABSTRACT:
The authors have described their experience in attempting to modularize Consul, a fault-tolerant system for managing replicated data. Their experience is that, while modularity is needed to simplify the design and implementation of such systems, protocol dependencies, both direct and indirect, impact the way in which protocols are designed and implemented. Also, the **authors** have **identified** certain **operating system features** that simplify such a design. Based on this experience, work is in progress to develop a new model for fault-tolerant protocols that will facilitate modularization.

DESCRIPTORS: COMMUNICATION PROTOCOLS; OPERATING SYSTEM--COMPUTERS; FILE MANAGEMENT; DISTRIBUTED COMPUTING; DISTRIBUTED PARAMETER SYSTEMS; ERROR RESILIENT SCHEME; MODULAR CONCEPT
IDENTIFIERS: DATENWIEDERHOLUNG--(VERTEILTE DV); CONSUL--(DATENVERWALTUNGSSYSTEM); Datenwiederholung; verteilte DV; Protokoll

12/5/68 (Item 2 from file: 95)
DIALOG(R)File 95:TEME-Technology & Management

00708350 E93093191080
Communication concepts of distributed operating systems
(Kommunikationskonzepte von verteilten Betriebssystemen)
Mandl, P; Schabernack, J
Univ. Koblenz-Landau, D
1993
Document type: Report Language: English
Record type: Abstract

ABSTRACT:
In recent years importance of distributed systems has increased rapidly and several distributed operating systems have been developed to manage the more complex hardware. In most approaches the **functionality** of **operating system** kernels is **reduced** and **functions** , like file service, are implemented in user level server processes to make systems more manageable. In these architectures services are requested by communication with servers rather than procedure calls. As a consequence design of communication plays an essential part in distributed **operating system design** . This paper surveys and compares the communication concepts of distributed operating systems Mach, Chorus and Helios based on a description of their central features, especially process concept and memory management.

DESCRIPTORS: OPERATING SYSTEM--COMPUTERS; DISTRIBUTED COMPUTING; COMMUNICATION; COMMUNICATION SYSTEMS; PARALLEL PROCESSING; MASSIVELY PARALLEL MACHINES; MEMORY MANAGEMENT; CLIENT SERVER SYSTEMS; OPEN SYSTEMS; DEFECT DETECTION; DISTRIBUTED OPERATING SYSTEMS
IDENTIFIERS: MACH; CHORUS; HELIOS; Kommunikationskonzept; verteiltes Betriebssystem

18/5/59 (Item 2 from file: 144)
DIALOG(R)File 144:Pascal
(c) 2004 INIST/CNRS. All rts. reserv.

11968427 PASCAL No.: 95-0149681
Objects to the rescue! or httpd : the next generation operating system
BLACK A P; WALPOLE J
Oregon graduate inst., dep. computer sci. eng., Portland OR 97291-1000, USA
Journal: Operating systems review, 1995, 29 (1) 91-95
ISSN: 0163-5980 CODEN: OSRED8 Availability: INIST-18399; 354000058094060120
No. of Refs.: 11 ref.
Document Type: P (Serial) ; A (Analytic)
Country of Publication: USA
Language: English

This position paper suggests that **object** -oriented **operating systems** may provide the means to meet the ever-growing demands of applications. As an example of a sucessful OOOS, we cite the http daemon. To support the contention that httpd is in fact an operating system, we observe that it implements uniform naming, persistent objects and an invocation meta-protocol, specifies and implements some useful objects, and provides a framework for extensibility. We also believe that the modularity that is characteristic of OO systems should provide a performance benefit rather than a penalty. Our ongoing work in the Synthetix project at OGI is exploring the possibilities for advanced optimization in such systems

English Descriptors: **Operating system ; Object** oriented; System performance; Optimization; **Modular** programming

French Descriptors: Systeme exploitation; Oriente objet; Performance systeme; Optimisation; Programmation modulaire

Classification Codes: 001D02B04

18/5/63 (Item 3 from file: 34)
DIALOG(R)File 34:SciSearch(R) Cited Ref Sci
(c) 2004 Inst for Sci Info. All rts. reserv.

01253241 Genuine Article#: GH771 Number of References: 30
Title: **SEMIAUTOMATIC** PROGRAM CONSTRUCTION **FROM SPECIFICATIONS USING LIBRARY MODULES**
Author(s): NISHIDA F; TAKAMATSU S; FUJITA Y; TANI T
Corporate Source: FUKUI INST TECHNOL,DEPT IND MANAGEMENT/FUKUI//JAPAN/; OITA UNIV,FAC ENGN,DEPT COMP SCI & INTELLIGENT SYST/OITA//JAPAN/; UNIV OSAKA PREFECTURE,FAC ENGN,DEPT ELECT ENGN/SAKAI/OSAKA 591/JAPAN/
Journal: IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, 1991, V17, N9, P853-871
Language: ENGLISH Document Type: ARTICLE
Geographic Location: JAPAN
Subfile: SciSearch; CC ENGI--Current Contents, Engineering, Technology & Applied Sciences
Journal Subject Category: ENGINEERING, ELECTRICAL & ELECTRONIC; COMPUTER APPLICATIONS & CYBERNETICS

Abstract: This paper describes a method of semi-automatic specification refinement and **program generation** using library **modules** . Users write their specifications, modify and rearrange them so that the specifications can be refined with the aid of the library modules. When a specification is given, a refinement system called MAPS searches for library modules applicable to the given specification, replaces the specification with a more detailed description written in the operation part of the modules, and converts the refined specification into a **program written** in a programming language designated by the user. Case-like expressions or pseudo-natural language expressions are used for describing user's specifications and specifications for library modules.